John P. Schnurer, SBN 185725
schnurer@fr.com
Cheng (Jack) C. Ko, SBN 244630
ko@fr.com
Ryan P. O'Connor, SBN 253596
oconnor@fr.com
Fish & Richardson P.C.
12390 El Camino Real
San Diego, CA 92130
Telephone: (858) 678-5070
Facsimile: (858) 678-5099

Attorney for Plaintiff
ASUSTEK Computer, Inc.

UNITED STATES DISTRICT COURT

SOUTHERN DISTRICT OF CALIFORNIA

ASUSTEK COMPUTER, INC.,

Plaintiff,

v.

INTERNATIONAL BUSINESS MACHINES CORPORATION,

Defendant.

Case No. '08 CV 602 JM WMc

COMPLAINT FOR PATENT INFRINGEMENT

DEMAND FOR JURY TRIAL

FILE BY FAX

Plaintiff ASUSTeK Computer, Inc. (hereinafter Plaintiff or "ASUS") hereby files this complaint against Defendant International Business Machines Corporation (hereinafter "IBM"), and alleges as follows:

PARTIES

1. Plaintiff ASUSTeK Computer, Inc. is a Taiwanese corporation with its principal place of business located at 4F No. 150 Li-Te Road, Peitou, Taipei 112, Taiwan, Republic of China (R.O.C.).

2. On information an belief, Defendant IBM is a New York corporation with its principal place of business located at New Orchard Road, Armonk, New York 10504.

JURISDICTION AND VENUE

3. This Court has subject matter jurisdiction over the cause of this action pleaded herein under 28 U.S.C. §§ 1331 and 1338(a) because the action concerns a federal question arising under the patent laws of the United States, including 35 U.S.C. § 271.

4. Venue is proper in this judicial district under 28 U.S.C. §§1391(b), (c) and 1400(b) because, among other reasons, IBM is subject to personal jurisdiction in this judicial district, has committed acts of infringement in this judicial district, and has a regular and established place of business in this judicial district.

5. Upon information and belief, IBM has placed infringing products into the stream of commerce by shipping those products into this judicial district or knowing that such products would be shipped into this judicial district.

FIRST CLAIM FOR RELIEF
INFRINGEMENT OF U.S PATENT NO. 6,041,346

6. The allegations of paragraphs 1-5 are incorporated herein by reference.

7. ASUS is the owner by assignment of all right, title, and interest in and to the United States Patent No. 6,041,346, entitled "Method and system for providing remote storage for an internet appliance" (hereinafter "the '346 patent"), which was duly and legally issued on March 21, 2000. A true and correct copy of the '346 patent is attached to this Complaint as Exhibit 1.

8. IBM has infringed and continues to infringe directly and/or indirectly, literally and/or under the doctrine of equivalents, one or more claims of the '346 patent under 35 U.S.C. § 271, by making, using, offering to sell, or selling within the United States, including this judicial district, or importing into the United States, storage area network products, software and/or components thereof.

9. IBM's unlawful infringement of the '346 patent, ASUS has suffered, and will continue to suffer, irreparable harm and damages.

10. IBM's past and continued acts of infringement have injured and damaged ASUS, and will continue to cause irreparable injury to ASUS unless and until enjoined by this Court.

SECOND CLAIM FOR RELIEF
INFRINGEMENT OF U.S PATENT NO. 7,103,765

11. The allegations of paragraphs 1-5 are incorporated herein by reference.

12. ASUS is the owner by assignment of all right, title, and interest in and to United States Patent No. 7,103,765, entitled "Method and system for providing a modulized server on board" (hereinafter "the '765 patent"), which was duly and legally issued on September 5, 2006. A true and correct copy of the '765 patent is attached to this Complaint as Exhibit 2.

13. IBM has infringed and continues to infringe directly and/or indirectly, literally and/or under the doctrine of equivalents, one or more claims of the '765 patent under 35 U.S.C. § 271, by making, using, offering to sell, or selling within the United States, including this judicial district, or importing into the United States, servers, software and/or components thereof.

14. IBM's unlawful infringement of the '765 patent, ASUS has suffered, and will continue to suffer, irreparable harm and damages.

PRAYER FOR RELIEF

WHEREFORE, ASUS prays that this Court enters judgment and provides relief as follows:

15. That IBM has infringed the '346 patent and the '765 patent;

(a) That IBM, and its officers, agents, servants, employees, and those in active concert or participation with them directly or indirectly, be enjoined from infringing the '346 patent and the '765 patent;

(b) That IBM be ordered to account for and pay to ASUS the damages resulting from IBM's infringement of the '346 patent and the '765 patent, together with interest and costs, and all other damages permitted by 35 U.S.C. § 284, including enhanced damages up to three times the amount of damages found or measured;

(c) That this action be adjudged an exceptional case and IBM be awarded its attorneys' fees, expenses and costs in this action pursuant to 35 U.S.C. § 285; and

(d) That ASUS be awarded such other equitable or legal relief as this Court deems just and proper under the circumstances.

DEMAND FOR JURY TRIAL

Defendant ASUS hereby demands a jury trial on all claims and issues in this action.

DATED: April 2, 2008

FISH & RICHARDSON P.C.

By: _____

John P. Schnurer
Cheng (Jack) C. Ko
Ryan P. O'Connor

Attorneys for Plaintiff,
ASUSTeK Computer, Inc.

NOTICE OF PARTY WITH FINANCIAL INTEREST

Pursuant to CivLR 40.2, the undersigned certifies that as of this date, there are no persons, associations of persons, firms, partnerships, corporations (including parent corporations) or other entities, other than the parties themselves, known by ASUS to have a financial interest in the subject matter in controversy or in a party to the proceeding.

DATED: April 2, 2008

FISH & RICHARDSON P.C.

By: 

John P. Schnurer
Cheng (Jack) C. Ko
Ryan P. O'Connor

Attorneys for Plaintiff,
ASUSTeK Computer, Inc.

EXHIBIT 1

US006041346A

United States Patent [19]

Chen et al.

[11] Patent Number: 6,041,346

[45] Date of Patent: Mar. 21, 2000

[54] METHOD AND SYSTEM FOR PROVIDING REMOTE STORAGE FOR AN INTERNET APPLIANCE

[75] Inventors: Ben W. Chen; Bo Xiong, both of Fremont, Calif.

[73] Assignee: Ateon Networks, Inc., Fremont, Calif.

[21] Appl. No.: 08/953,029

[22] Filed: Oct. 17, 1997

[51] Int. Cl.[7] G06F 13/00

[52] U.S. Cl. 709/218

[58] Field of Search 395/200.47, 200.48, 395/200.49; 709/217, 218, 219; 711/147

[56] References Cited

U.S. PATENT DOCUMENTS

5,754,771	5/1998	Epperson et al.	709/203
5,802,299	9/1998	Logan et al.	709/218
5,826,242	10/1998	Montulli	705/27
5,832,505	11/1998	Kasso et al.	707/104
5,889,949	3/1999	Charles	709/214

Primary Examiner—Kenneth Coulter
Attorney, Agent, or Firm—Sawyer Law Group LLP

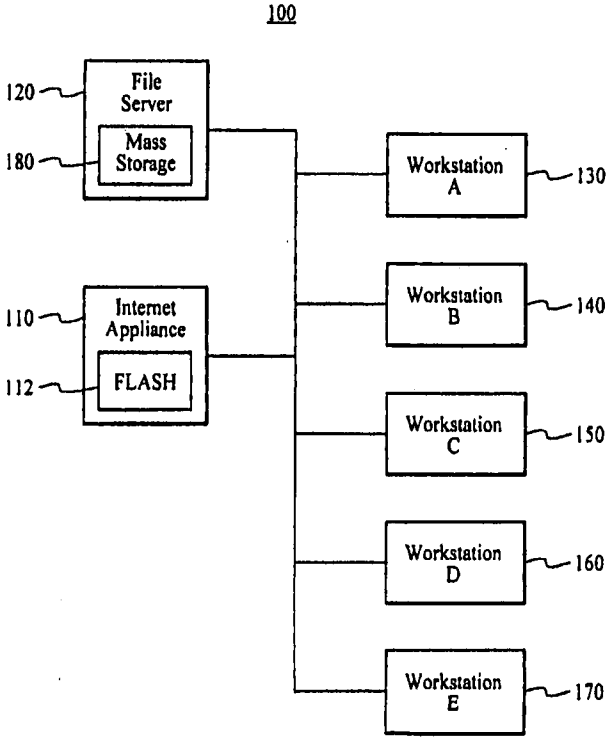
[57] ABSTRACT

The present invention provides a method and system for providing access to a remote network. In a preferred embodiment, a system and method in accordance with the present invention allows a local host on the private network to transparently access the remote network and permit multiple users in the local network to simultaneously access the remote network. This access is based upon the requirements of an application utilizing a mass storage device within the private network. This system allows for user security on the private network.

In one aspect, the method and system allow a private network to access a remote network. The private network has a plurality of components. At least one the plurality of components has a mass storage device. In this aspect, the method and system comprise providing an internet appliance and providing a portion of the mass storage device of the at least one of the plurality of components. The portion of the mass storage device is shareable with and mapped to the internet appliance. In a second aspect, the method and system comprise providing an internet appliance and providing a first component having a mass storage device. A portion of the mass storage device is shareable with and mapped to the internet appliance.

According to the system and method disclosed herein, the present invention provides an internet appliance having remote storage.

40 Claims, 4 Drawing Sheets



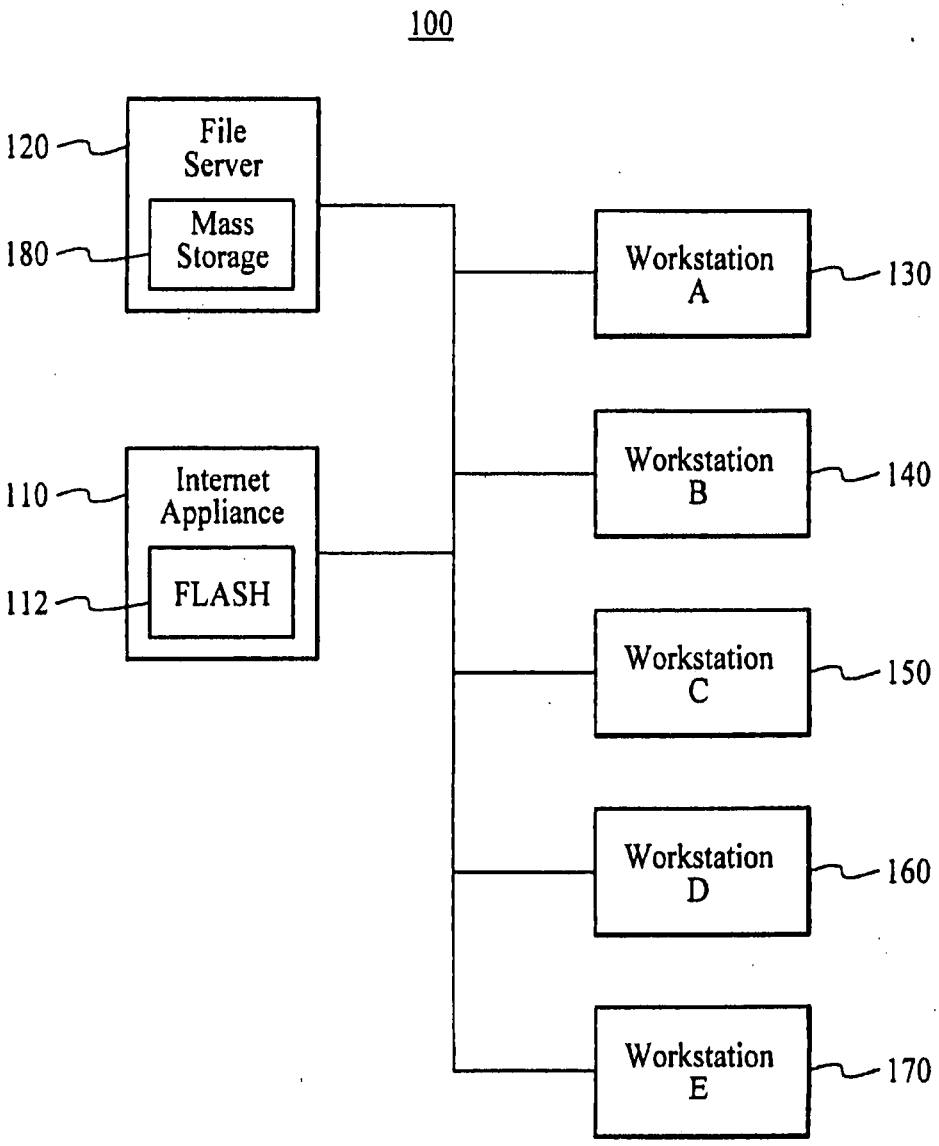

100
120
File Server
180
Mass Storage
110
Internet Appliance
112
FLASH
Workstation A
130
Workstation B
140
Workstation C
150
Workstation D
160
Workstation E
170

FIG. 1

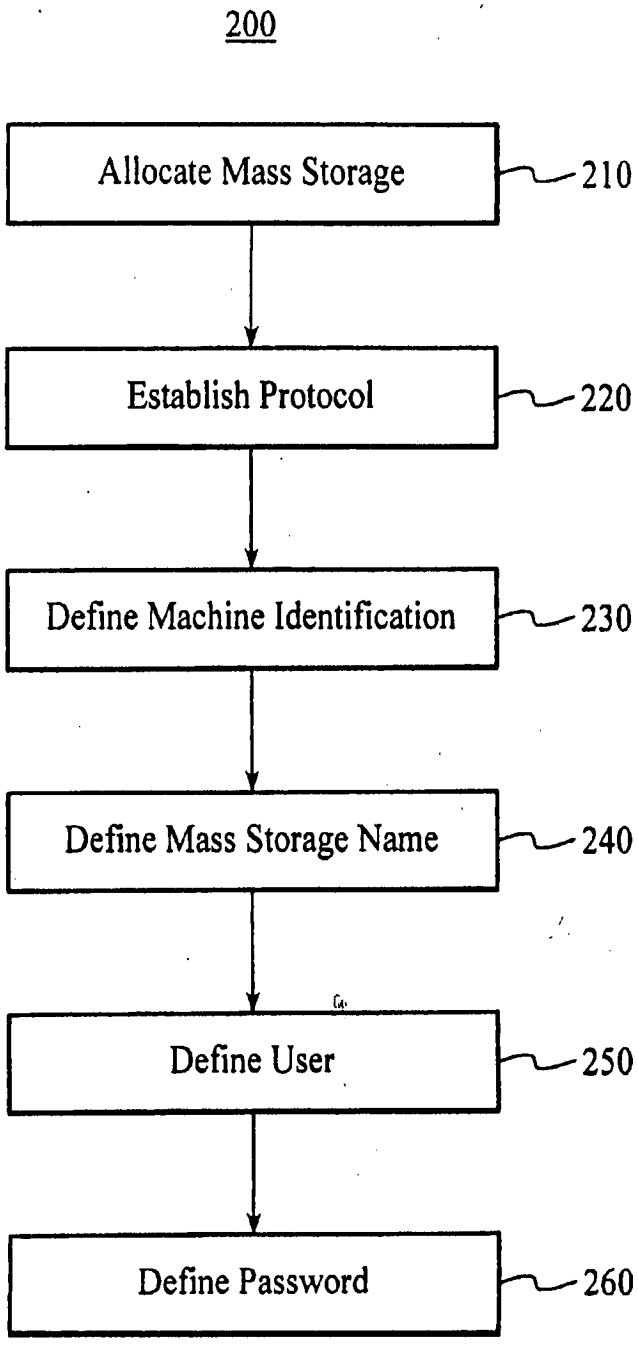

200
Allocate Mass Storage
210
Establish Protocol
220
Define Machine Identification
230
Define Mass Storage Name
240
Define User
250
Define Password
260

FIG. 2

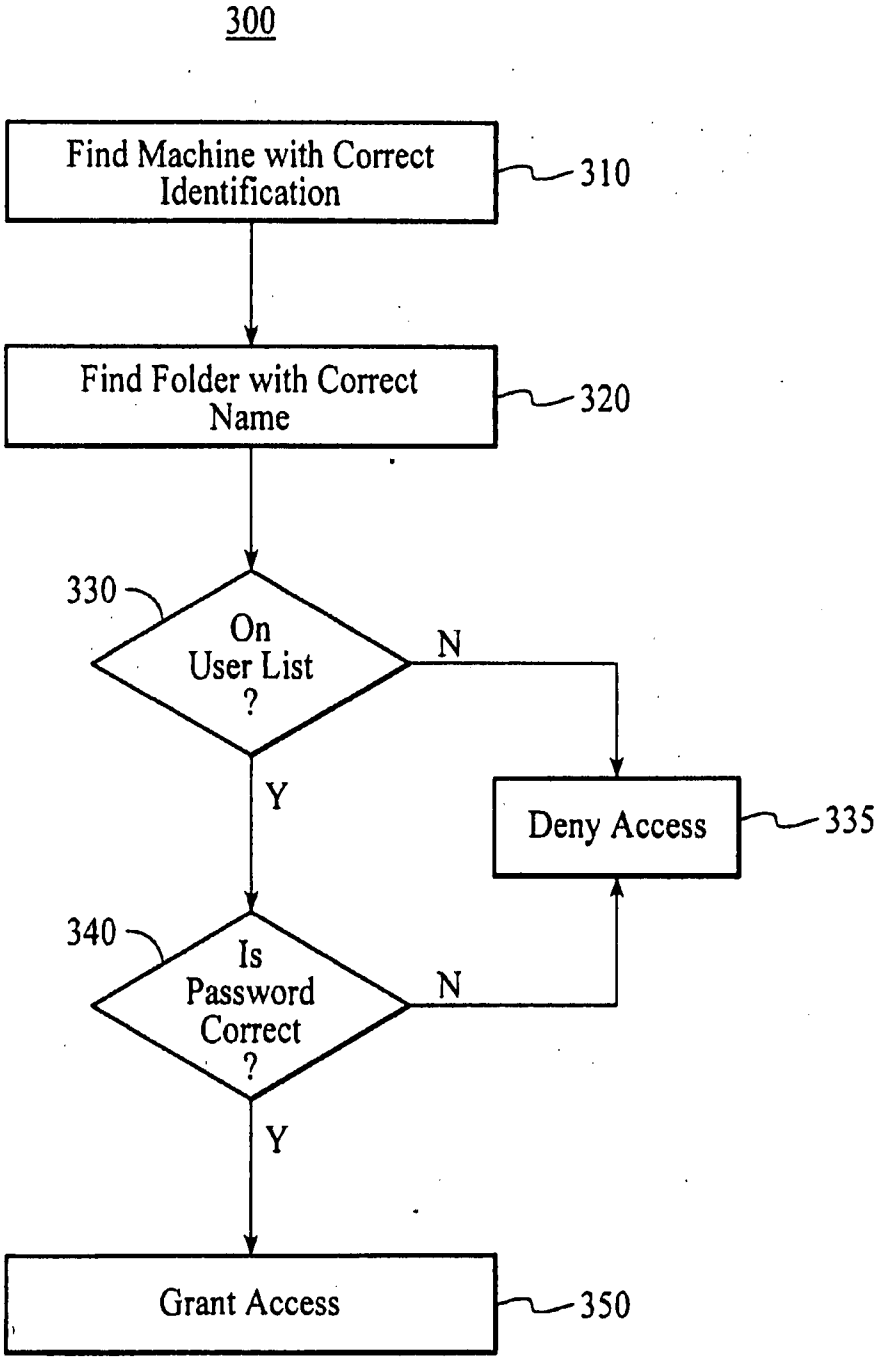

300
Find Machine with Correct Identification
310
Find Folder with Correct Name
320
330
On User List ?
N
Y
Deny Access
335
340
Is Password Correct ?
N
Y
Grant Access
350

FIG. 3

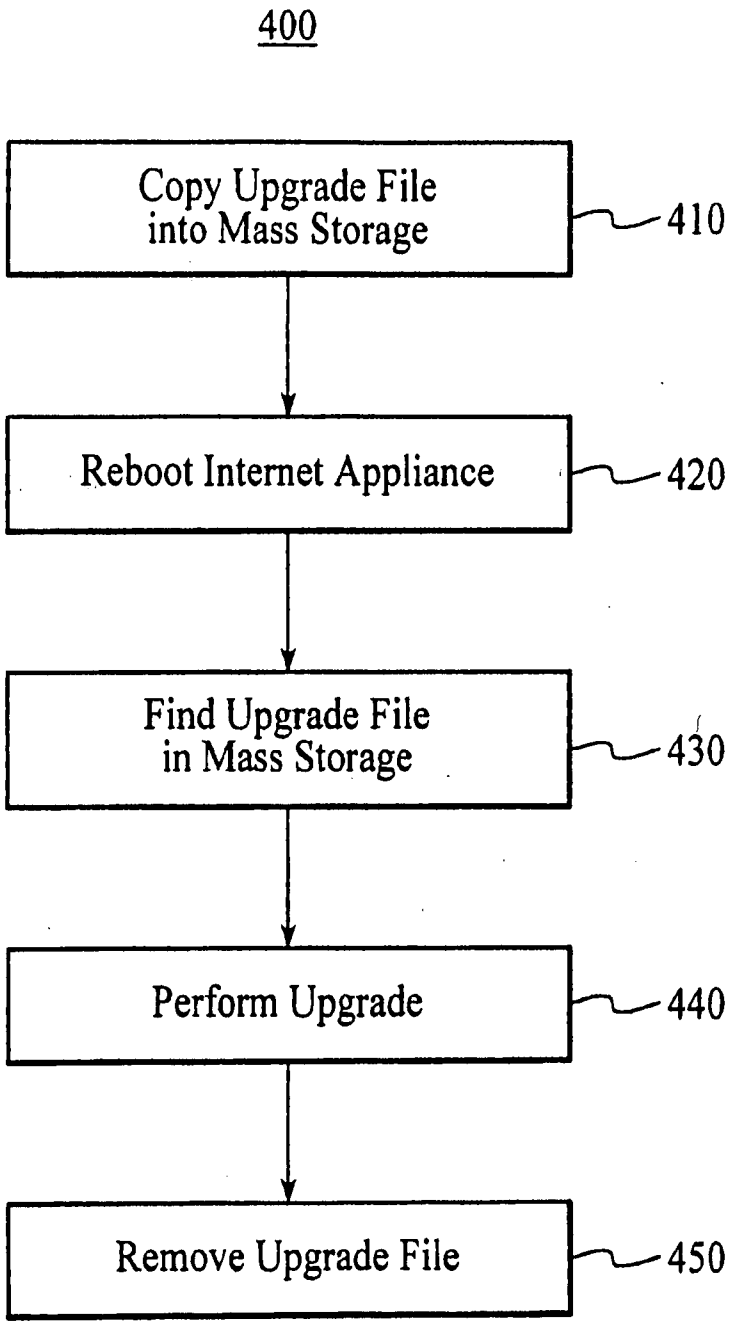

400
Copy Upgrade File into Mass Storage
410
Reboot Internet Appliance
420
Find Upgrade File in Mass Storage
430
Perform Upgrade
440
Remove Upgrade File
450

FIG. 4

METHOD AND SYSTEM FOR PROVIDING REMOTE STORAGE FOR AN INTERNET APPLIANCE

FIELD OF THE INVENTION

The present invention relates to internet appliances and more particularly to a method and system for providing an internet appliance having remote storage.

BACKGROUND OF THE INVENTION

Currently many users are interested in communicating with remote systems. An internet appliance is a device that helps provide access to the remote system, such as a remote network. For example, an internet appliance can aid in gaining access to such a remote network including but not limited to an online service such as COMPUSERVE, AMERICA ONLINE, an internet service provider, or a subset of any of these services.

Conventional internet appliances typically have limited functionality. For example, a conventional router only has sufficient memory to perform the required connection and routing functions. Such conventional internet appliances cannot provide built-in applications and perform such functions as electronic mail to users in the network because these functions require large storage devices.

Some conventional internet appliances have sufficient additional internal memory to provide additional functions. Such conventional internet appliances are known as intelligent internet appliances. Conventional intelligent internet appliances have many levels of functionality. However, these internet appliances are typically complete personal computer systems. Conventional intelligent network appliances having only minimal internal memory provide some additional functions. Other conventional intelligent internet appliances, such as those having an internal hard drive, have added storage and, therefore, added functionality. In order to take advantage of the applications relating to the internet, it may be desirable to use intelligent internet appliance to provide the application to the network users. In order to do so, the internal memory of the conventional intelligent internet appliance is used.

For example, routers may be used to couple workstations of a network to the internet. Some conventional intelligent routers have sufficient internal storage to provide applications. Such conventional intelligent routers can store an operating system and other controlling software, as well as applications such as electronic mail.

Applications such as electronic mail may require a large amount of memory. The actual quantity of memory required for electronic mail varies depending on several factors, including the number of users on a network. Typically, the amount of space required can vary from several megabytes to well over one hundred megabytes. A typical intelligent internet appliance should have at least approximately one hundred megabytes of storage. In order to reduce costs, this internal storage is typically an internal hard drive.

Although conventional intelligent internet appliances having an internal hard drive are capable of providing applications such as electronic mail, use of a hard drive creates other difficulties. For example, a hard drive in the conventional internet appliance adversely affects the reliability of the conventional internet appliance. If, for example, the hard drive fails, it may be difficult to remove the hard drive from the conventional internet appliance. In addition, a back up utility is required when utilizing a hard drive to ensure reliability. This back up utility adds additional cost and complexity to the system. The hard drive also may not limit which users can obtain access to the hard drive. As a result, the security of the hard drive and any application residing on the hard drive may be compromised. Accordingly, while all these features are achievable they can prohibitively add to the cost and complexity of the appliance.

Accordingly, what is needed is a system and method for providing an internet appliance which can be used to provide storage for applications or other uses. The present invention addresses such a need.

SUMMARY OF THE INVENTION

The present invention provides a method and system for providing access to a remote network. In a preferred embodiment, a system and method in accordance with the present invention allows a local host on the private network to transparently access the remote network and permit multiple users in the local network to simultaneously access the remote network. This access is based upon the requirements of an application utilizing a mass storage device within the private network. This system allows for user security on the private network.

In one aspect, the method and system allow a private network to access a remote network. The private network has a plurality of components. At least one the plurality of components has a mass storage device. In this aspect, the method and system comprise providing an internet appliance and providing a portion of the mass storage device of the at least one of the plurality of components. The portion of the mass storage device is shareable with and mapped to the internet appliance. In a second aspect, the method and system comprise providing an internet appliance and providing a first component having a mass storage device. A portion of the mass storage device is shareable with and mapped to the internet appliance.

According to the system and method disclosed herein, the present invention provides an internet appliance having remote storage.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of one embodiment of a network including an internet-appliance having remote storage in accordance with the method and system.

FIG. 2 is a is a flow chart depicting a method for providing remote storage for an internet appliance in accordance with the method and system.

FIG. 3 is a is a flow chart depicting a method for accessing the remote storage for an internet appliance in accordance with the method and system.

FIG. 4 is a flow chart depicting a method for upgrading the internet appliance in accordance with the method and system.

DETAILED DESCRIPTION OF THE INVENTION

The present invention relates to an improvement in internet appliances. As used herein, an internet appliance is a device which aids in providing access to a remote system, such as a remote network. The following description is presented to enable one of ordinary skill in the art to make and use the invention and is provided in the context of a patent application and its requirements. Various modifications to the preferred embodiment will be readily apparent to

those skilled in the art and the generic principles herein may be applied to other embodiments. Thus, the present invention is not intended to be limited to the embodiment shown but is to be accorded the widest scope consistent with the principles and features described herein.

Conventional internet appliances, such as routers, typically have only sufficient memory to provide the required functions, such as connection to the remote system and routing between components of the network. Other intelligent internet appliances have enough additional memory to provide additional functions. As used herein, an internal mass storage is the additional memory available in intelligent internet appliances. When sufficient internal mass storage is provided for conventional intelligent internet appliances, such internet appliances can provide applications, such as electronic mail ("email").

Most conventional internet appliances have no internal mass storage for two reasons. First, the amount of internal mass storage required for applications such as email is indeterminate. For example, the amount of memory required for email may vary based on the number of users. Instead of determining the requisite amount of internal mass storage, most manufacturers simply provide no internal mass storage. The second reason why most conventional internet appliances have no internal mass storage is that internal mass storage is very costly. Hard drives are commercially available typically in a minimum size. Oftentimes this minimum size is much greater than the amount of storage required for the application. In addition, the cost of hard drives is high. For applications requiring less memory, other types of memory, such as FLASH memory, can be utilized. However, the cost of FLASH memory is high. As a result, for many applications the cost of FLASH memory is prohibitive. Thus, most conventional internet appliances have no internal mass storage. Consequently, one of ordinary skill in the art will recognize that most conventional internet appliances have limited functionality due to the limited amount of internal mass storage.

Conventional intelligent internet appliances having mass storage, such as a hard drive, for example, approximately one hundred MB, may be capable of providing additional features. Although conventional intelligent internet appliances having an internal hard drive are capable of providing applications such as electronic mail, use of a hard drive creates other difficulties. For example, a hard drive in the conventional internet appliance adversely affects the reliability of the conventional internet appliance. If, for example, the hard drive fails, it may be difficult to remove the hard drive from the conventional internet appliance. This back up utility adds additional cost and complexity to the system. The hard drive also may not limit which users can obtain access to the hard drive. As a result, the security of the hard drive and any application residing on the hard drive may be compromised. Accordingly, while all these features are achievable they can prohibitively add to the cost and complexity of the internet appliance.

The present invention provides for a method and system for providing an internet appliance having internal mass storage that can be used to provide applications. The method and system allocate a portion of the internal mass storage in another system on the network, such as a workstation, and map the mass storage in the other system on the network to the internet appliance. Thus, the method and system include allowing the internet appliance to share the mass storage with the other component. The present invention will be described in terms of a router having mass storage used to run an application such as email. However, one of ordinary skill in the art will readily recognize that this method and system will operate effectively for other types of internet appliances and other applications.

To more particularly illustrate the method and system in accordance with the present invention, refer now to FIG. **1** depicting a block diagram of one embodiment of a network **10** employing such a system. The network **10** includes a file server **120**, workstation A **130** through workstation E **170**, and an internet appliance **110** in accordance with the method and system. In a preferred embodiment, the internet appliance **110** is used to provide email to the workstation A **130** through workstation E **170**. However, nothing prevents the method and system from providing a different application. In addition, although the network **10** is depicted as including only five workstations, nothing prevents the method and system from being used on a network having another number of workstations.

In one embodiment, the internet appliance **110** includes a memory **112** having limited storage capability. In a preferred embodiment, the memory **112** is a FLASH memory with limited storage capacity. Also in a preferred embodiment, the memory **112** is capable of storing an operating system and other controlling software. In one embodiment, the operating system and other controlling software are completely storable in the memory **112**. However, in another embodiment, the operating system and other controlling software are stored in a mass storage device **180**, discussed below. In such a case, the memory **112** includes the software required to boot up the internet appliance **110** and use the software on the mass storage device **180**. Because the memory **112** includes only limited storage capacity, the memory **112** is inexpensive.

A portion of the mass storage device **180** of the file server **120** is allocated for use in providing an application. Thus, the mass storage device of the file server **120** is the memory for the internet appliance **110**. Although the mass storage device **180** is shown as being located in the file server **120**, the mass storage device **180** can be in any component having sufficient memory including workstation A **130** through workstation E **170** and the file server **120**. In one embodiment, the mass storage device **180** stores an application to be provided to the network **100** by the internet appliance **110** but does not store the operating system. In another embodiment, the mass storage device **180** includes the operating system and controlling software in addition to the application.

The mass storage device **180** is shareable and mapped to the internet appliance **110**. As a result, the internet appliance **110** can use the mass storage device **180**. In a preferred embodiment, the application is electronic mail and the mass storage device **180** is a folder. Note that nothing prevents the method and system from allocating the mass storage device **180** in another component including the file server **120** or another workstation.

FIG. **2** depicts a flow chart of one method **200** for establishing a link between the mass storage device **180** and the internet appliance **110**. The memory is allocated in the remote location, such as file server **120** via step **210**. The protocol to be used is then established in step **210**. The protocol establishes the manner in which another component, such as the internet appliance **110**, can communicate with the mass storage device **180**. The machine on which the mass storage device **180** resides is then identified in step **230**. This allows the internet appliance **110** to query the appropriate component, file server **120**, to access the mass storage device **180**. In the network **10** depicted in FIG.

1, the machine identified in step **230** is file server **120**. The mass storage device **180** is then given a name in step **240**. In a preferred embodiment, the mass storage device **180** is a folder named virtual mailbox.

The users of the mass storage device **180** are then defined in step **250**. For example, the internet appliance **110** is defined as a user in the step **250**. The password required to access the mass storage device **180** is then provided via step **260**. The application residing in the mass storage device **180** is secure because only identified users can access the mass storage device **180** and a password is required to be provided by the internet appliance **110** in order access the mass storage device **180**. Once the protocols have been established, the folder named, and the user and password provided, the mass storage device **180** is shareable.

FIG. **3** depicts a method **300** for accessing the mass storage device **180**. When the internet appliance **110**, or another component of the network **100**, attempts to access the mass storage device **180**, then via step **310** the internet appliance **110** will search for the component with the proper identity as defined in step **230**. In step **320** the internet appliance **110** then finds the appropriate folder in the component using the name of the mass storage device **180** provided in step **240**. It is determined via step **330** that the internet appliance **110** is a named user. If another component which is not a named user attempts to access the mass storage device **180**, then the component will be denied access via step **335**. If the internet appliance is a named component, then it must be determined if the password is correct, via step **340**. If the correct password is not provided, access to the mass storage device **180** is denied. If the password is correct, authorized users can then access the mass storage device **180**.

Using the mass storage device **180**, the internet appliance **110** can provide applications to workstation A **130** through workstation E **170**. In a preferred embodiment, the mass storage device **180** and the internet appliance **110** provide email capability to the workstation A **130** through the workstation E **170**. In one embodiment, the capacity of the mass storage device **180** can be any memory available on the file server **120**.

Because the internet appliance **110** does not require a substantial amount of internal memory, the internet appliance **110** can be provided at a much lower cost than a conventional intelligent internet appliance. Because the mass storage device **180** used by the internet appliance **110** is not physically located on the internet appliance **110**, the mass storage device **180** can be more easily removed for servicing, thereby improving reliability. Because the method **200** provides a list of users and passwords which are based on the network security infrastructure, the mass storage device **180** is also more secure than the internal memory for a conventional internet appliance.

In one embodiment, the internet appliance **110** is easily upgradable. The internet appliance **110** can be provided with a mechanism for recognizing an upgrade file. Refer now to FIG. **4** depicting one embodiment of a method **400** for upgrading the internet appliance **110**. An upgrade file is placed in a recognizable location, such as the mass storage device **180**, via step **410**. In one embodiment, the placement of the upgrade file in the mass storage device **180** can be accomplished by visiting a website, downloading the upgrade file, and moving the upgrade file into the removable memory. The internet appliance **110** is then rebooted via step **420**. Upon rebooting, the internet appliance **110** finds the upgrade file, performs the upgrade, and removes the upgrade file, via steps **430**, **440**, and **450**, respectively. In one embodiment, the installation step **440** includes decrypting the file as well as performing the upgrade. Thus, in addition to providing applications such as email to the workstation A **130** through the workstation E **170**, the internet appliance **110** is also easily upgradable.

A method and system has been disclosed for providing a mass storage device for an internet appliance which may allow the internet appliance to provide applications. Although the present invention has been described in accordance with the embodiments shown, one of ordinary skill in the art will readily recognize that there could be variations to the embodiments and those variations would be within the spirit and scope of the present invention. Accordingly, many modifications may be made by one of ordinary skill in the art without departing from the spirit and scope of the appended claims.

What is claimed is:

1. A system for providing access to a remote network comprising:

an internet appliance; and

a first component coupled with the internet appliance and having a mass storage device, a portion of the mass storage device being sharable with and mapped to the internet appliance;

wherein the internet appliance is capable of using any component that is coupled with the internet appliance and that includes the mass storage device as the first component.

2. The system of claim **1** further comprising an application residing on the internet appliance, the internet appliance being capable of providing the application.

3. The system of claim **2** wherein the portion of the mass storage device is accessible only by a user identified on a user list and having a password, the user list including the internet appliance; and wherein the internet appliance further includes the password.

4. The system of claim **3** wherein the internet appliance further comprises an internal memory.

5. The system of claim **4** wherein the internal memory further comprises a FLASH memory.

6. The system of claim **5** wherein the internet appliance is further automatically upgradable via the remote network.

7. The system of claim **1** wherein the internet appliance is further automatically upgradable via the remote network.

8. The system of claim **1** wherein the first component is a workstation.

9. The system of claim **1** wherein the internet appliance is a router.

10. A system for allowing a private network to access a remote network, the private network having a plurality of components, at least one of the plurality of components having a mass storage device, the system comprising:

an internet appliance coupled with the plurality of components; and

a portion of the mass storage device of the at least one of the plurality of components, the portion of the mass storage device being sharable and mapped to the internet appliance;

wherein the internet appliance is capable of using any of the plurality of components having the mass storage device and coupled with the internet appliance as the at least one of the plurality of components.

11. The system of claim **1** further comprising an application residing on the internet appliance, the internet appliance being capable of providing the application.

12. The system of claim 11 wherein the application further comprises an electronic mail application.

13. The system of claim 12 wherein the portion of the mass storage device is accessible only by a user identified on a user list and having a password, the user list including the internet appliance; and wherein the internet appliance further includes the password.

14. The system of claim 1 wherein the internet appliance further comprises a router having an internal memory.

15. The system of claim 14 wherein the internal memory further comprises a FLASH memory.

16. The system of claim 1 wherein the internet appliance is further automatically upgradable via the remote network.

17. The system of claim 1 wherein the at least one of the plurality of components further comprises a workstation.

18. The system of claim 1 wherein the at least one of the plurality of components further comprises a file server.

19. The system of claim 16 wherein the high speed online service provider is an internet service provider.

20. A method for providing access to a remote network comprising the steps of:

providing an internet appliance;

providing a component coupled with the internet appliance and having a mass storage device;

allocating a portion of the mass storage device for use by the internet appliance; and

making the portion of the mass storage device shareable with and mapped to the internet appliance;

wherein the internet appliance is capable of using any component that is coupled with the internet appliance and that has the mass storage device as the component.

21. The method of claim 20 wherein the step of making the mass storage device shareable with and mapped to the internet appliance further comprises the steps of:

providing a protocol, the protocol for allowing communication between the internet appliance and the portion of the mass storage device;

providing an identity of the component; and

providing a name of the portion of the mass storage device.

22. The method of claim 21 wherein the internet appliance further includes a password; and wherein the step of making the mass storage device shareable with and mapped to the internet appliance further comprises the steps of:

providing a user list, the user list including the internet appliance; and

providing a password, the portion of the mass storage device being accessible only by a user having a password and included on the user list.

23. The method of claim 20 further comprising the step of:

providing an, the application being stored in the internet appliance, the internet appliance being capable of providing the application.

24. The method of claim 23 wherein the step of providing the internet appliance further comprises the step of;

providing an internal memory.

25. The method of claim 24 wherein the internal mass storage device further comprises a FLASH memory.

26. The method of claim 25 wherein the step of providing the internet appliance further comprises the step of:

providing an upgradable internet appliance capable of being upgraded by accessing the remote network.

27. The method of claim 20 wherein the step of providing the internet appliance further comprises the step of:

providing an upgradable internet appliance capable of being upgraded by accessing the remote network.

28. The method of claim 20 wherein the component is a workstation.

29. The method of claim 20 wherein the internet appliance is a router.

30. A method for providing a private network access to a remote network, the private network including a plurality of components, at least one of the plurality of components including a mass storage device; the method comprising the steps of:

providing an internet appliance coupled with the plurality of components;

allocating a portion of the mass storage device of the at least one of the plurality of components for use by the internet appliance; and

making the portion of the mass storage device shareable with and mapped to the internet appliance;

wherein the internet appliance is capable of using any of the plurality of components having the mass storage device and coupled with the internet appliance as the at least one of the plurality of components.

31. The method of claim 30 wherein the step of making the portion of the mass storage device shareable and mapped to the internet appliance further comprises the steps of:

providing a protocol the protocol for allowing communication between the internet appliance and the portion of the mass storage device;

providing an identity of the at least one of the plurality of components; and

providing a name of the portion of the mass storage device.

32. The method of claim 31 wherein the internet appliance further includes a password; and wherein the step of making the mass storage device shareable and mapped to the internet appliance further comprises the steps of:

providing a user list, the user list including the internet appliance; and

providing a password, the portion of the mass storage device being accessible only by a user having a password and included on the user list.

33. The method of claim 30 further comprising the step of:

providing an, the application being stored in the internet appliance, the internet appliance being capable of providing the application.

34. The method of claim 33 wherein the step of providing an application further comprises the step of providing an electronic mail application.

35. The method of claim 30 wherein the at least one of the plurality of components further comprises a workstation.

36. The method of claim 30 wherein the at least one of the plurality of components further comprises a file server.

37. The method of claim 30 wherein the step of providing the internet appliance further comprises the step of:

providing a router with an internal memory.

38. The method of claim 37 wherein the internal mass storage device further comprises a FLASH memory.

39. The method of claim 30 wherein the step of providing the internet appliance further comprises the step of:

providing an upgradable internet appliance capable of being upgraded by accessing the remote network.

40. The method of claim 39 wherein the remote network is an internet service provider.

* * * * *

EXHIBIT 2

US007103765B2

(12) **United States Patent**
Chen

(10) **Patent No.: US 7,103,765 B2**
(45) **Date of Patent: Sep. 5, 2006**

(54) **METHOD AND SYSTEM FOR PROVIDING A MODULIZED SERVER ON BOARD**

(76) Inventor: **Ben Wei Chen**, 1400 Tolteca Ct., Fremont, Alameda, CA (US) 94539

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **10/002,652**

(22) Filed: **Oct. 19, 2001**

(65) **Prior Publication Data**

US 2003/0061474 A1 Mar. 27, 2003

Related U.S. Application Data

(60) Provisional application No. 60/324,900, filed on Sep. 25, 2001.

(51) **Int. Cl.**
G06F 1/24 (2006.01)
G06F 12/00 (2006.01)

(52) **U.S. Cl.** **713/2**; 713/1; 713/100; 711/100

(58) **Field of Classification Search** 713/1, 713/2, 100; 711/100; 361/684, 685
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,751,714	A	5/1998	Albano et al.	
5,889,970	A	3/1999	Horan et al.	
5,974,547	A	10/1999	Klimenko	
5,987,536	A	11/1999	Johnson et al.	
6,115,755	A	9/2000	Krishan	
6,256,732	B1 *	7/2001	Cromer et al.	713/2
6,282,643	B1 *	8/2001	Cromer et al.	713/2
6,408,333	B1	6/2002	St. Clair	

FOREIGN PATENT DOCUMENTS

EP	1113646	A1	7/2001

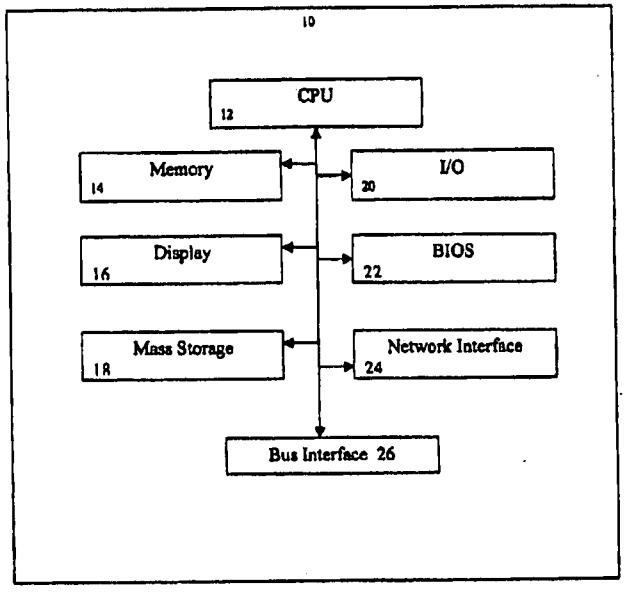
* cited by examiner

Primary Examiner—Rehana Perveen
(74) *Attorney, Agent, or Firm*—Sawyer Law Group LLP

(57) **ABSTRACT**

A method and system for providing a modulized server-on-a-board is disclosed. The server-on-a-board is installed on a computing device. The method and system include providing bus interface logic, providing local control BIOS, a flash memory and a set of control button connectors, light emitting diodes (LED) connectors and a liquid crystal display (LCD) connector. The local control BIOS is coupled with the bus interface logic and the flash memory. The bus interface logic interacts with the computing device and allows computing device to detect the server board. The local control BIOS boots up the server and prepares the computing device for use as the server. The flash memory stores a server image for the server, which is provided to the computing device using the local control BIOS. The control button connectors allow the server to be turned on, shut down gracefully, or restored to its initial state, by a single press of buttons connected to these connectors. The LED and LCD connectors allow the system status to be displayed or shown.

24 Claims, 11 Drawing Sheets



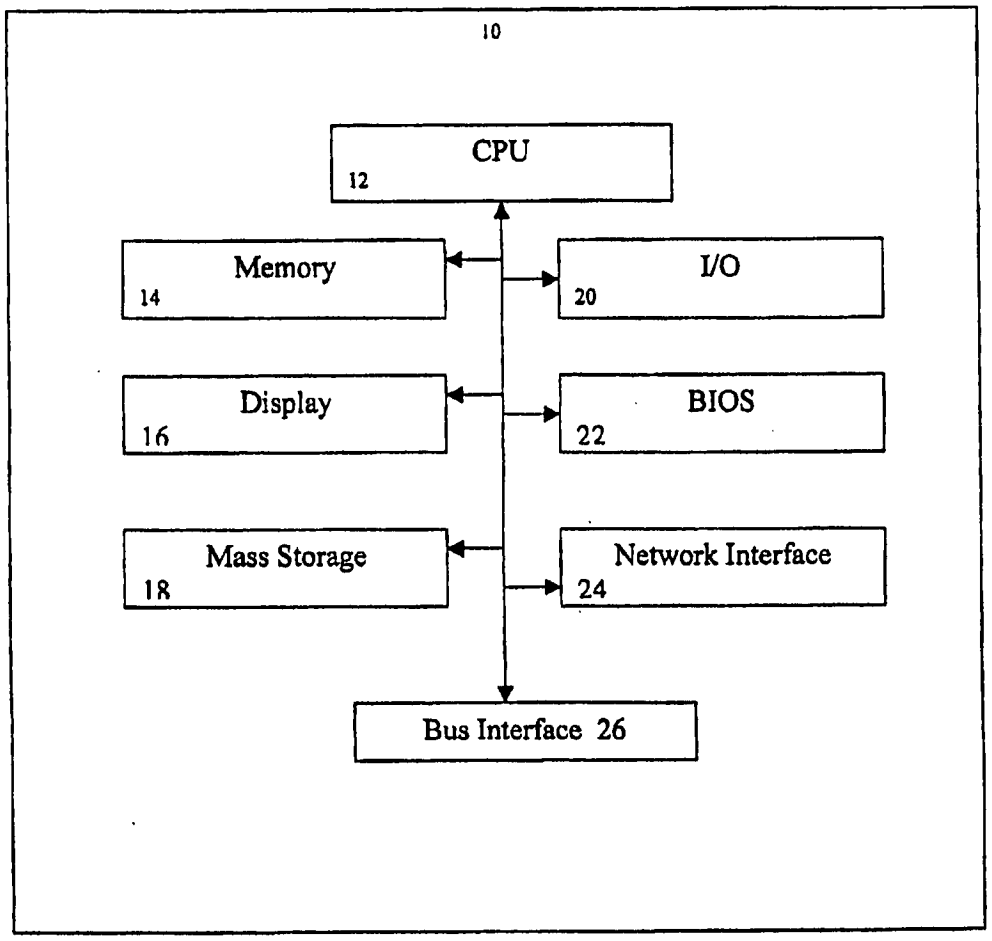

10
CPU
12
Memory
14
I/O
20
Display
16
BIOS
22
Mass Storage
18
Network Interface
24
Bus Interface 26

Figure 1

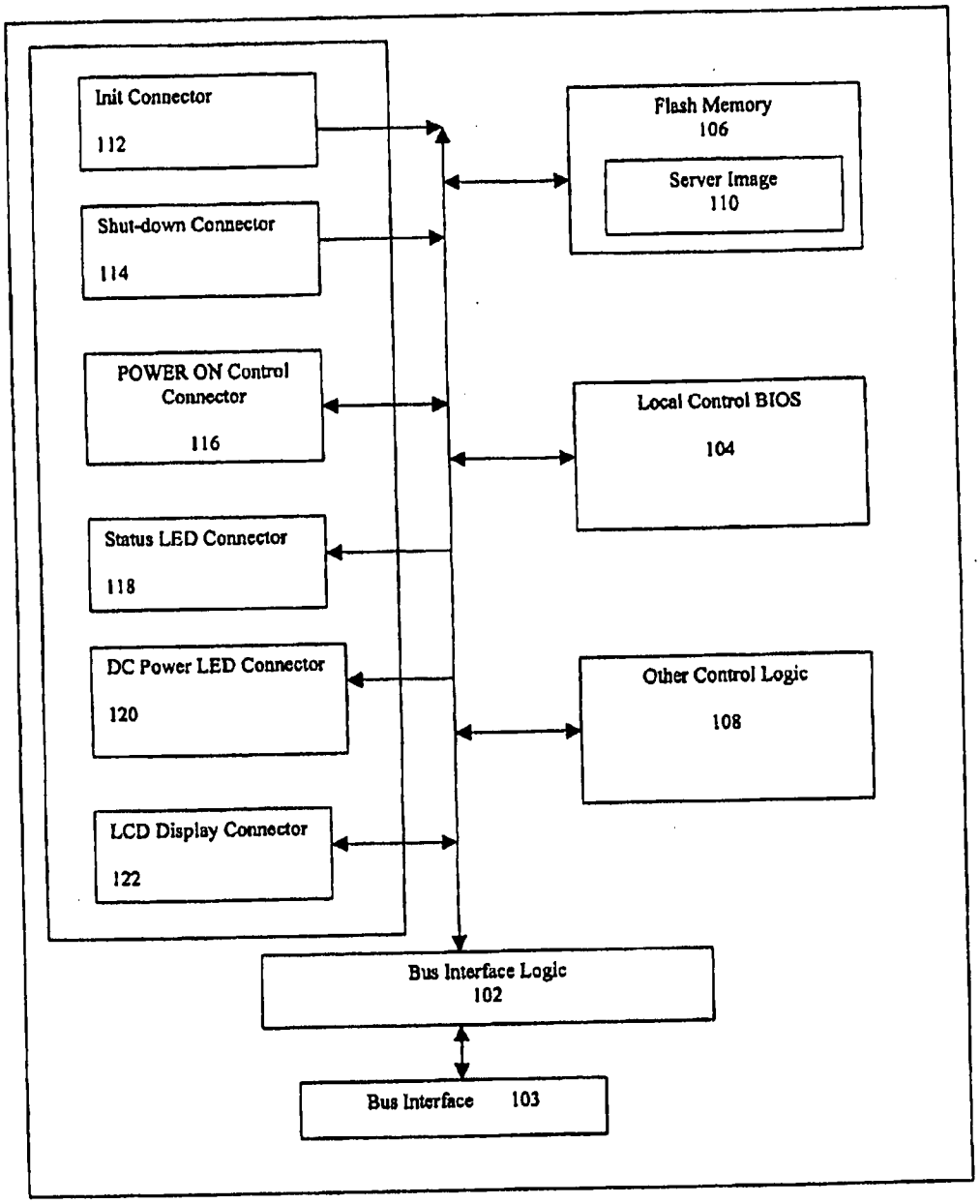

Init Connector
112
Shut-down Connector
114
POWER ON Control Connector
116
Status LED Connector
118
DC Power LED Connector
120
LCD Display Connector
122
Flash Memory
106
Server Image
110
Local Control BIOS
104
Other Control Logic
108
Bus Interface Logic
102
Bus Interface 103

Figure 2

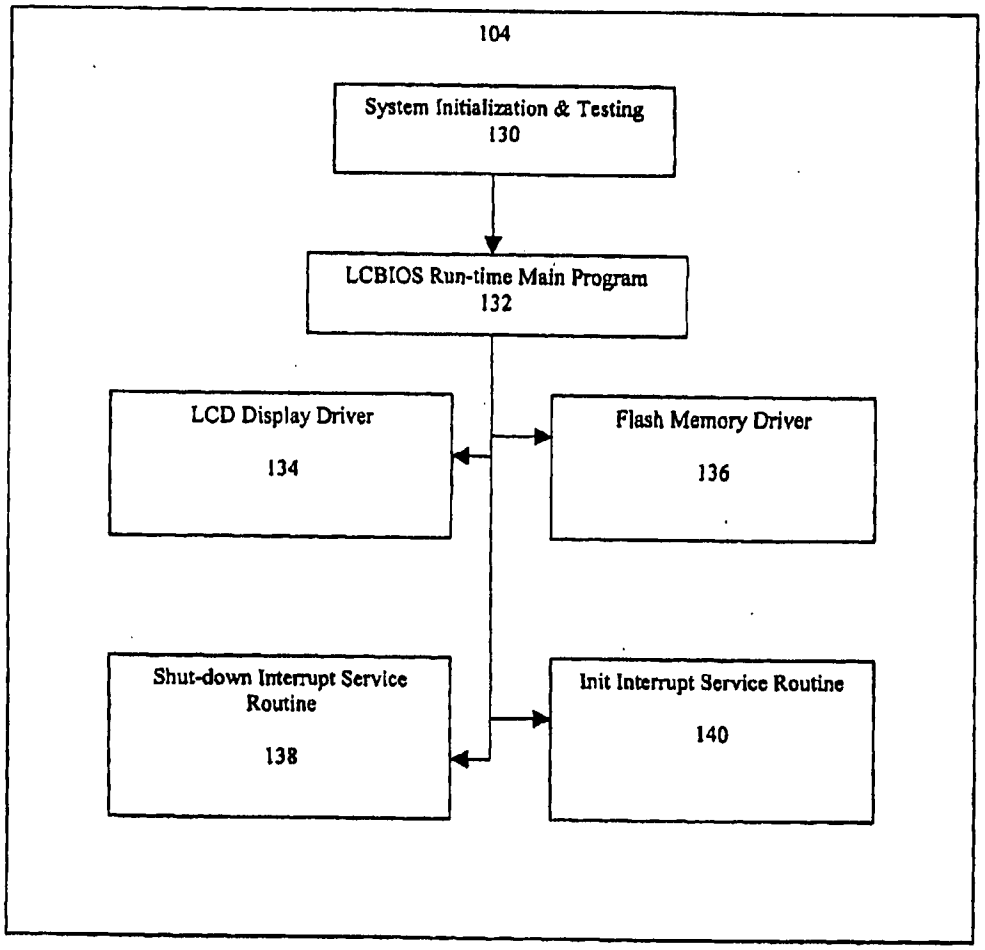

104
System Initialization & Testing
130
LCBIOS Run-time Main Program
132
LCD Display Driver
134
Flash Memory Driver
136
Shut-down Interrupt Service Routine
138
Init Interrupt Service Routine
140

Figure 3

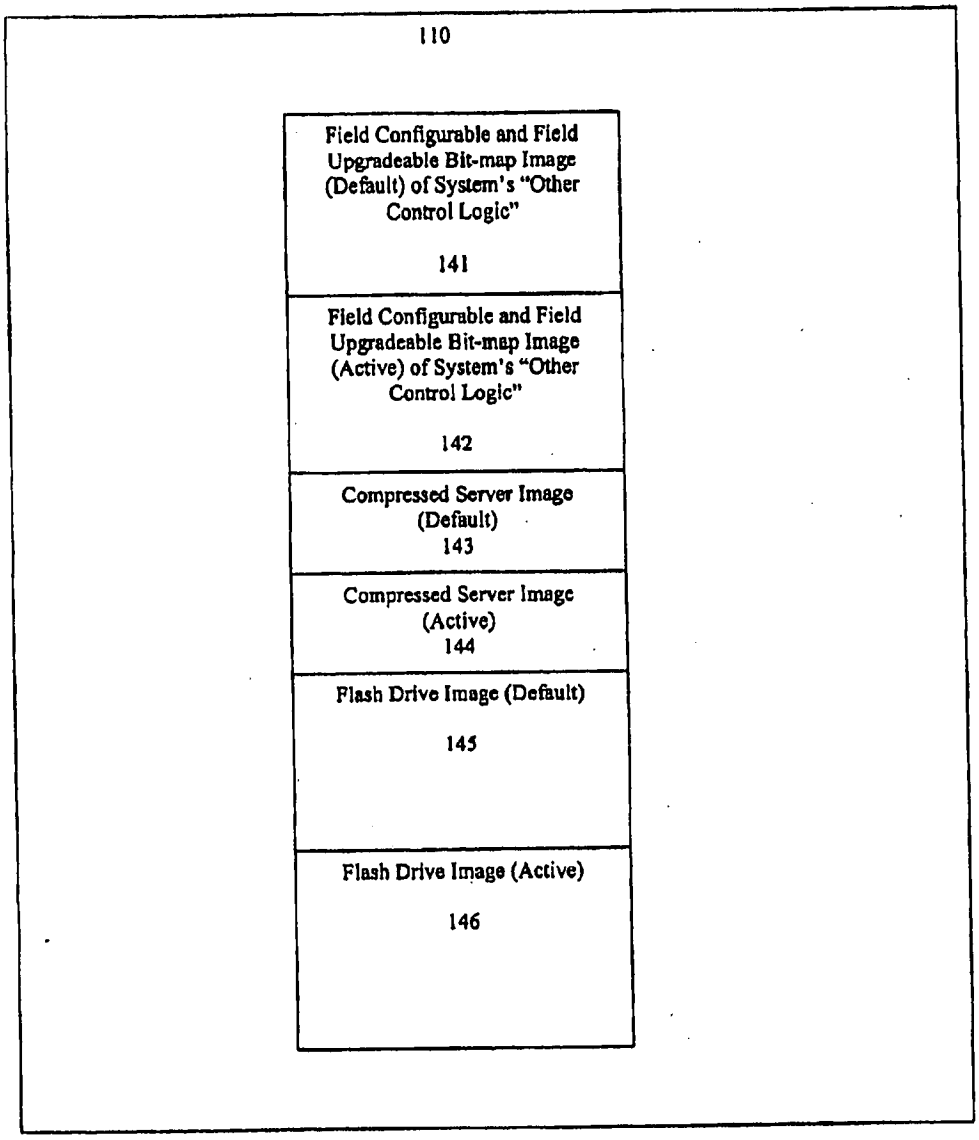

110
Field Configurable and Field Upgradeable Bit-map Image (Default) of System's "Other Control Logic"
141
Field Configurable and Field Upgradeable Bit-map Image (Active) of System's "Other Control Logic"
142
Compressed Server Image (Default)
143
Compressed Server Image (Active)
144
Flash Drive Image (Default)
145
Flash Drive Image (Active)
146

Figure 4

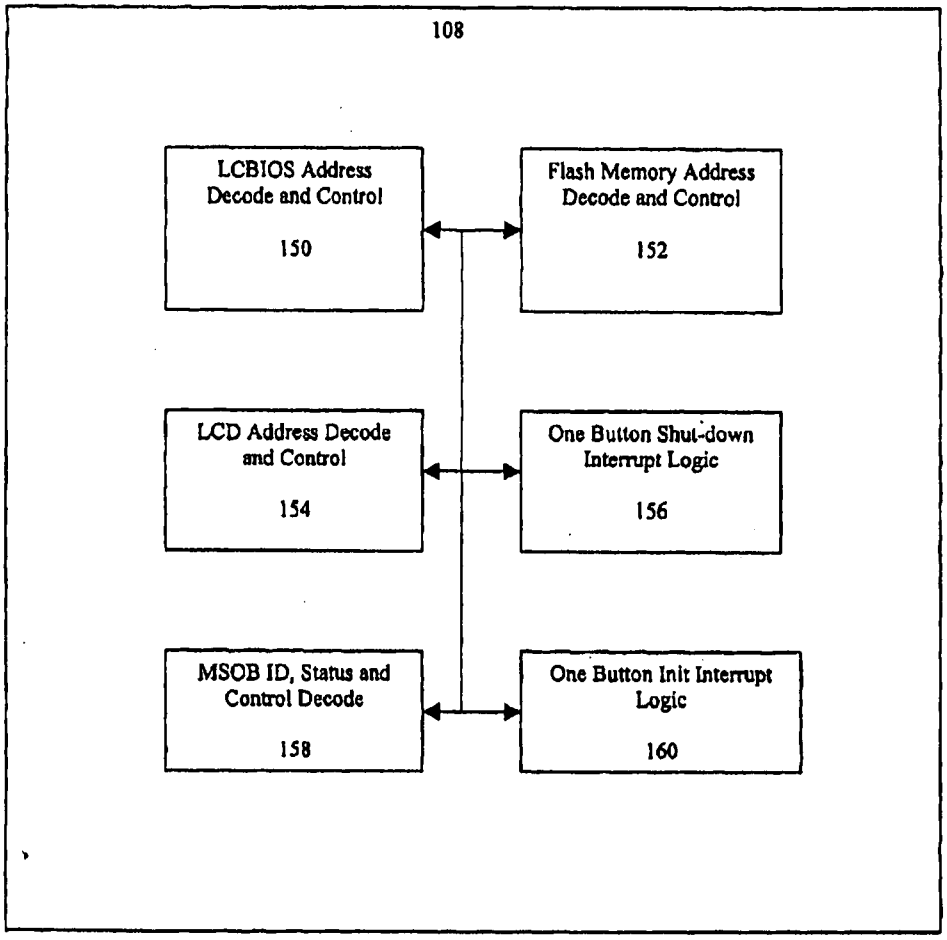

108
LCBIOS Address Decode and Control
150
Flash Memory Address Decode and Control
152
LCD Address Decode and Control
154
One Button Shut-down Interrupt Logic
156
MSOB ID, Status and Control Decode
158
One Button Init Interrupt Logic
160

Figure 5

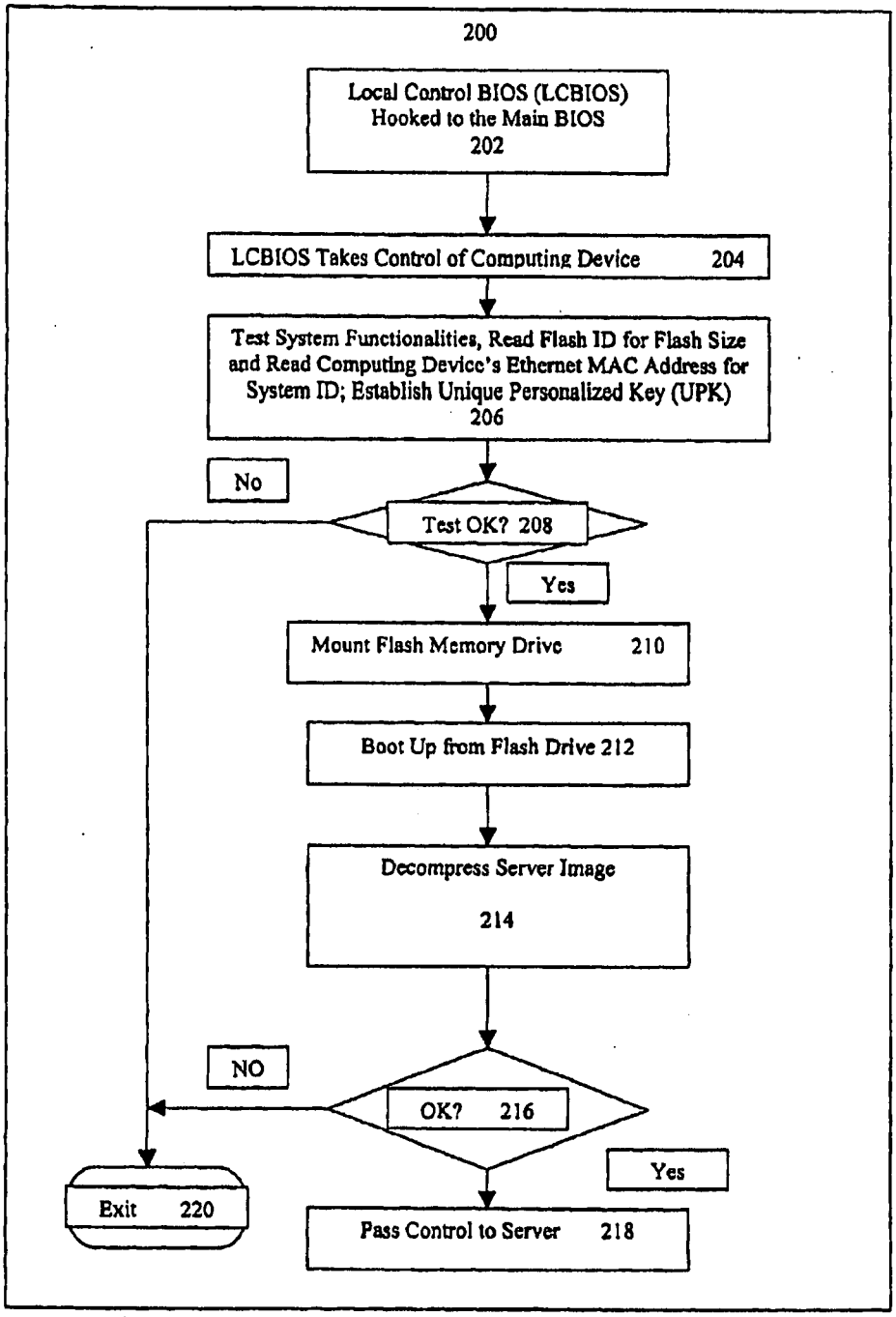

200
Local Control BIOS (LCBIOS) Hooked to the Main BIOS 202
LCBIOS Takes Control of Computing Device 204
Test System Functionalities, Read Flash ID for Flash Size and Read Computing Device's Ethernet MAC Address for System ID; Establish Unique Personalized Key (UPK) 206
No
Test OK? 208
Yes
Mount Flash Memory Drive 210
Boot Up from Flash Drive 212
Decompress Server Image 214
NO
OK? 216
Yes
Exit 220
Pass Control to Server 218

Figure 6

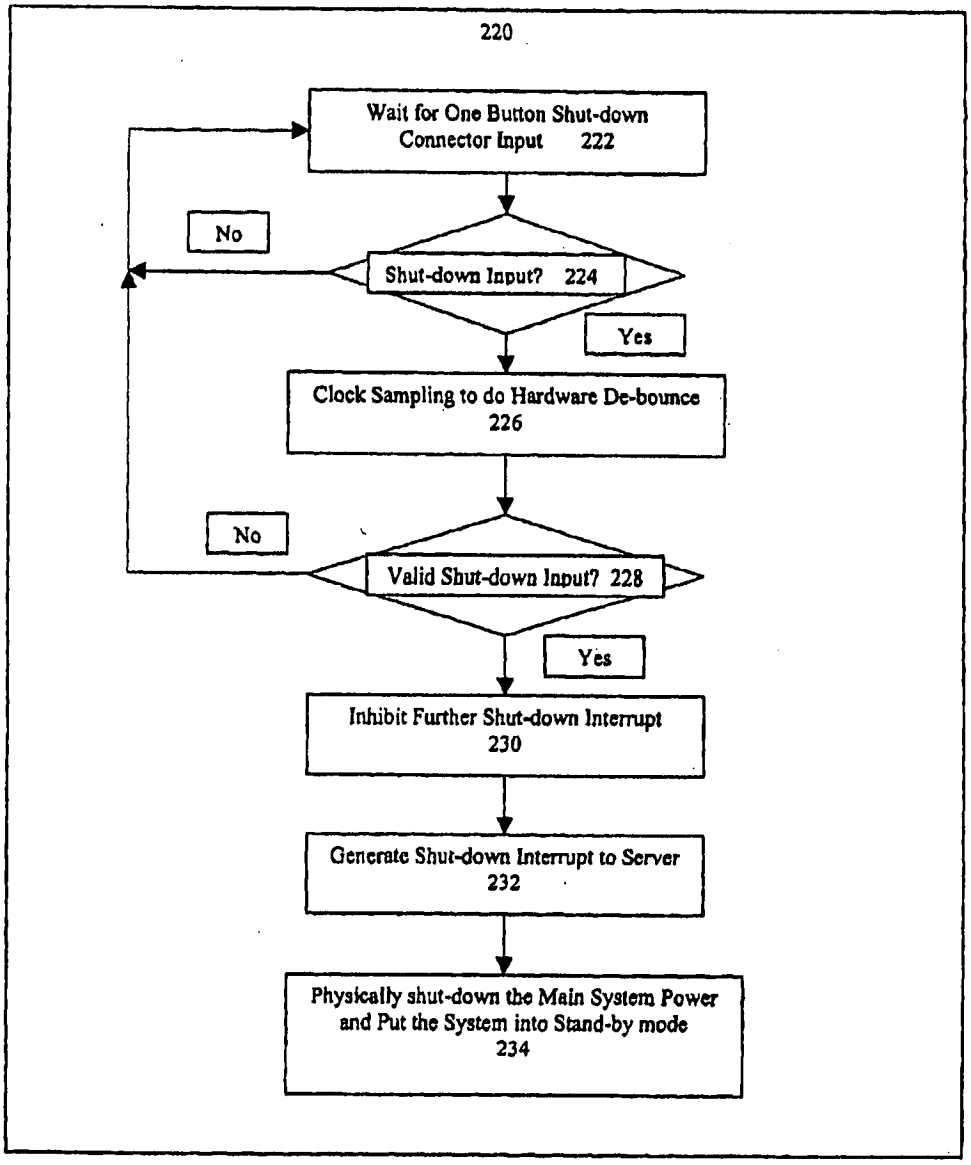

220
Wait for One Button Shut-down Connector Input 222
No
Shut-down Input? 224
Yes
Clock Sampling to do Hardware De-bounce 226
No
Valid Shut-down Input? 228
Yes
Inhibit Further Shut-down Interrupt 230
Generate Shut-down Interrupt to Server 232
Physically shut-down the Main System Power and Put the System into Stand-by mode 234

Figure 7

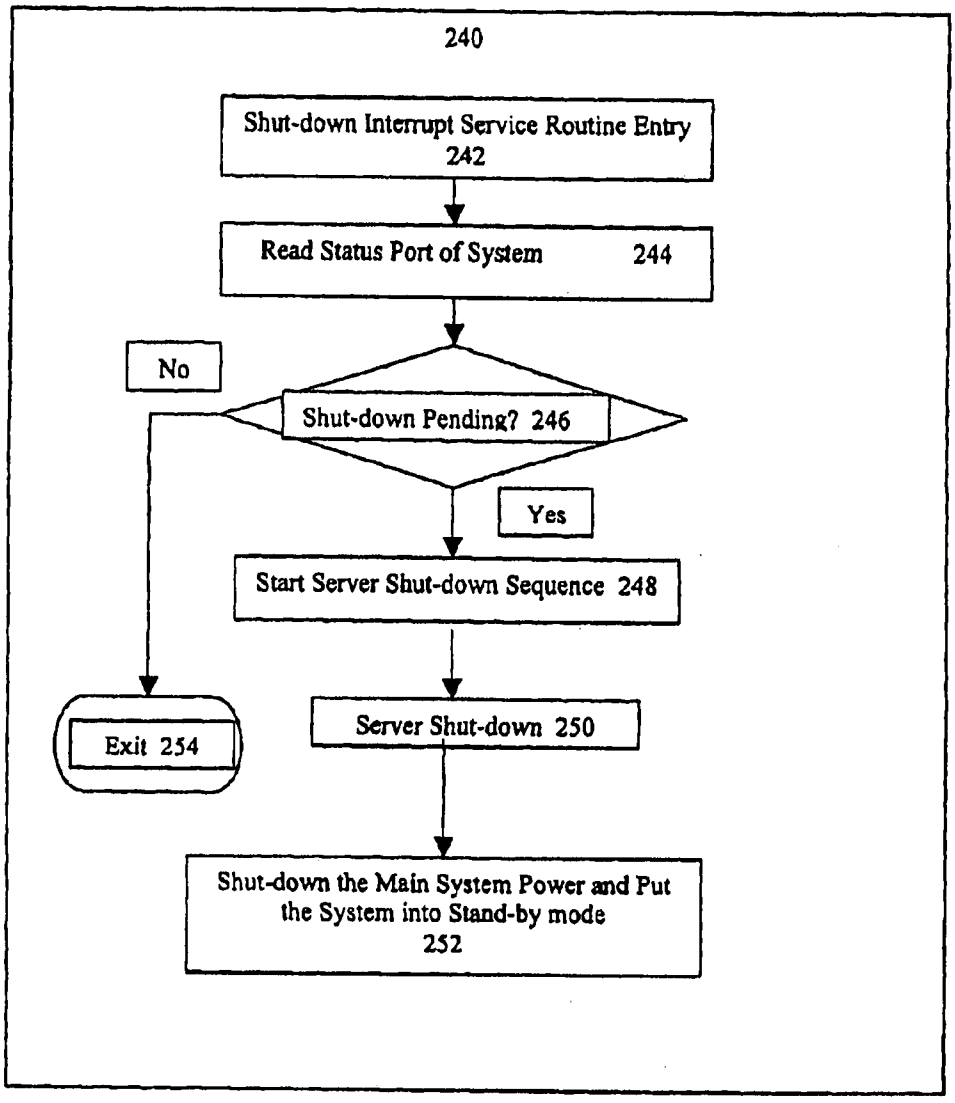

240
Shut-down Interrupt Service Routine Entry
242
Read Status Port of System 244
No
Shut-down Pending? 246
Yes
Start Server Shut-down Sequence 248
Exit 254
Server Shut-down 250
Shut-down the Main System Power and Put the System into Stand-by mode
252

Figure 8

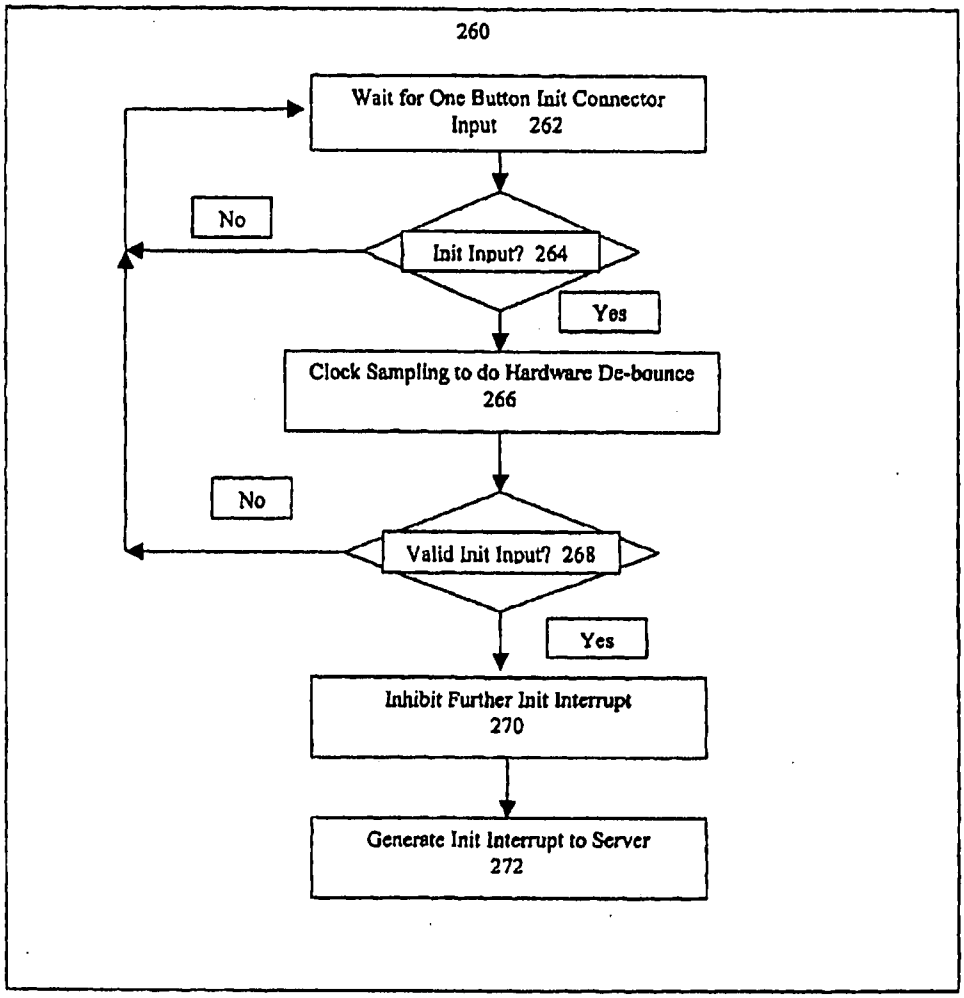

260
Wait for One Button Init Connector Input 262
No
Init Input? 264
Yes
Clock Sampling to do Hardware De-bounce 266
No
Valid Init Input? 268
Yes
Inhibit Further Init Interrupt 270
Generate Init Interrupt to Server 272

Figure 9

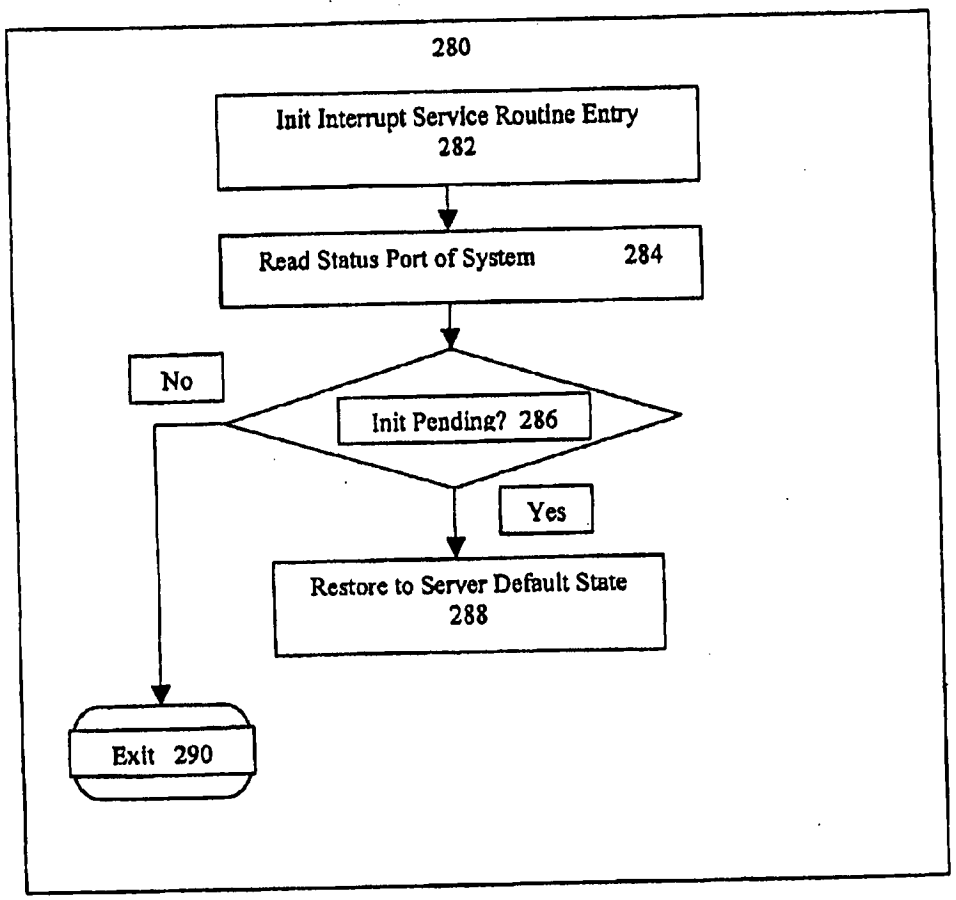

280
Init Interrupt Service Routine Entry
282
Read Status Port of System
284
No
Init Pending? 286
Yes
Restore to Server Default State
288
Exit 290

Figure 10

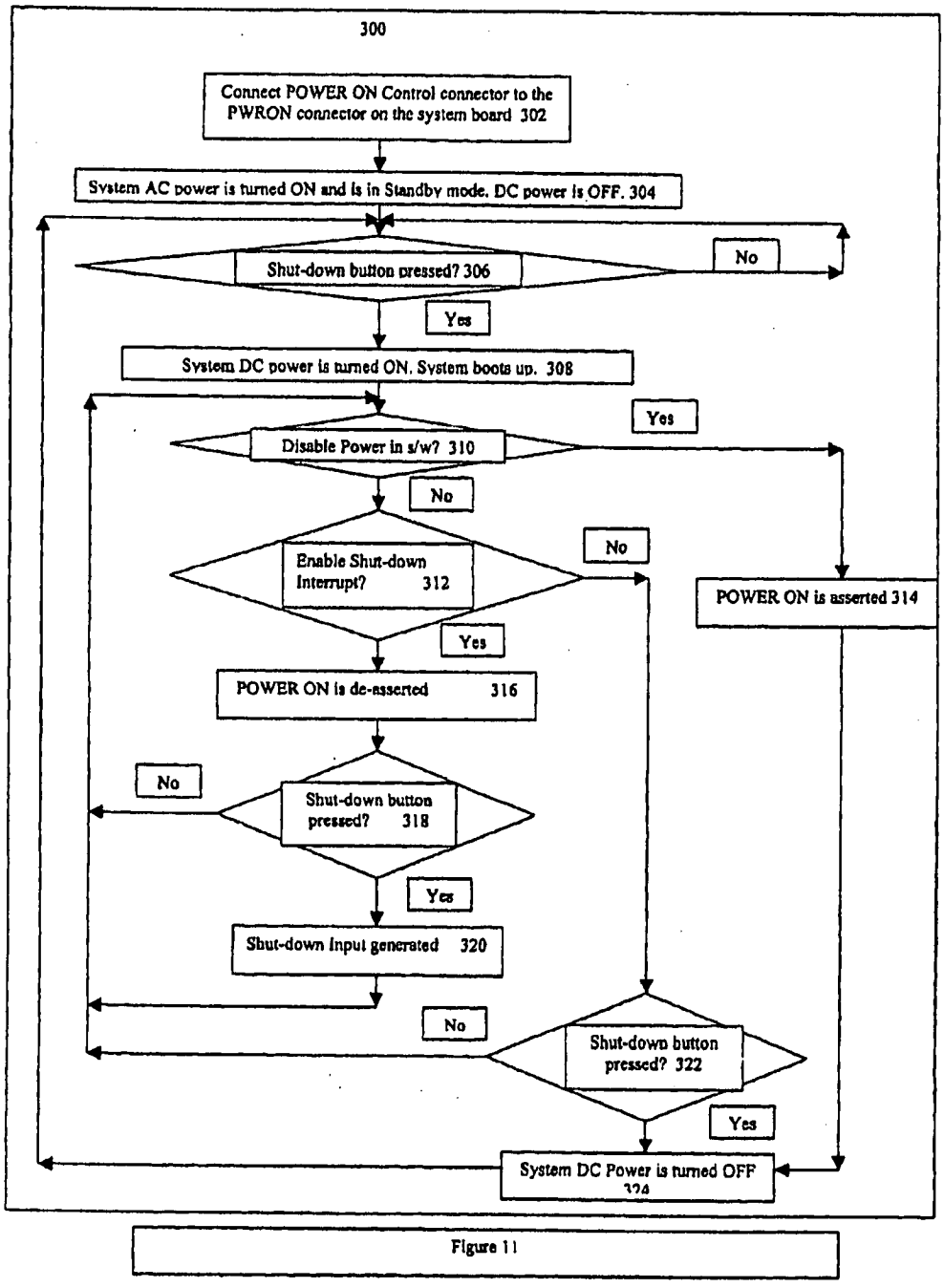

300
Connect POWER ON Control connector to the PWRON connector on the system board 302
System AC power is turned ON and is in Standby mode. DC power is OFF. 304
Shut-down button pressed? 306
No
Yes
System DC power is turned ON. System boots up. 308
Yes
Disable Power in s/w? 310
No
No
Enable Shut-down Interrupt? 312
POWER ON is asserted 314
Yes
POWER ON is de-asserted 316
No
Shut-down button pressed? 318
Yes
Shut-down Input generated 320
No
Shut-down button pressed? 322
Yes
System DC Power is turned OFF

Figure 11

METHOD AND SYSTEM FOR PROVIDING A MODULIZED SERVER ON BOARD

CROSS-REFERENCE TO RELATED APPLICATION

This application is claiming under 35 USC 119(c) the benefit of provisional patent Application Ser. No. 60/324,900 filed Sep. 25, 2001.

FIELD OF THE INVENTION

The present invention relates to computer systems, and more particularly to a method and system for providing a server on a generalized computing device.

BACKGROUND OF THE INVENTION

FIG. **1** depicts a generalized computing device ("computing device") **10**. The computing device **10** includes at least a CPU **12** and an optional mass storage **18**, such as a hard disk. The computing device **10** may also include other features. The computing device depicted in FIG. **1** also includes a memory **14** such as a flash memory, a display **16**, an input/output device **20** such as a keyboard, BIOS **22**, a network interface **24** and a bus interface **26**. Communication to a network (not shown) is carried out through the network interface **24**. Similarly, communication to any attached devices (not shown) can be carried out via the bus interface **26**. For example, the bus interface **26** could include interfaces for PCI, USB, SCSI, IDE, Infiniband or other connectors.

The computing device **10** is capable of performing a variety of functions. It is often desirable to utilize the computing device **10** as a server. A server would include additional hardware and/or software that allows the server to serve multiple users. Thus, the server would allow multiple users to share resources, such as printers or the optional mass storage **18** of the computing device **10**.

There are a number of conventional methods for allowing the computing device **10** to be used as a server. In general, these conventional methods involve obtaining server software and installing the software on the computing device **10**. The user must then manually set up the desired functions for the server. Alternatively, the computing device **10** could be specially built to function as a server. In either case, ensuring that the computing device **10** can function as a server is expensive. For example, obtaining and installing server software on the computing device **10** or specially building the computing device **10** may cost between $500 and $5,000. Moreover, installing the software and tailoring the system to provide the desired individual functions requires a substantial investment of time on the part of the user.

Accordingly, what is needed is a system and method for cheaply and easily allowing the computing device **10** to be used as a server. The present invention addresses such a need.

SUMMARY OF THE INVENTION

The present invention provides a method and system for providing a server on a computing device. The computing device includes at least a processor and an optional mass storage device. The method and system comprise providing bus interface logic, providing local control BIOS, a flash memory and, preferably, a set of control button connectors, light emitting diodes (LED) connectors and a liquid crystal display (LCD) connector. The local control BIOS is coupled with the bus interface logic and the memory. The bus interface logic interacts with the computing device and allows the computing device to detect the system. The local control BIOS boots up the server and prepares the computing device for use as the server. The memory stores a server image for the server, which is provided to the computing device using the local control BIOS. The control button connectors allow the server to be turned on, shut down gracefully, or restored to its initial state, by a single press of buttons connected to these connectors. The LED and LCD connectors allow the system status to be displayed or shown.

According to the system and method disclosed herein, the present invention provides an inexpensive, easy to use mechanism for allowing the computing device to be used as a server.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram of a conventional computing device.

FIG. **2** is a high level block diagram of a system in accordance with the present invention for allowing the computing device to be used as a server.

FIG. **3** is a block diagram of one embodiment of the BIOS of the system in accordance with the present invention for allowing the computing device to be used as a server.

FIG. **4** is a diagram of one embodiment of the image of the server stored in the memory of the system in accordance with the present invention for allowing the computing device to be used as a server.

FIG. **5** is a more detailed block diagram of one embodiment of the other control logic in the system in accordance with the present invention for allowing the computing device to be used as a server.

FIG. **6** is a flow chart of one embodiment of a method in accordance with the present invention for utilizing the system in accordance with the present invention to allow the computing device to be used as a server.

FIG. **7** is a flow chart of one embodiment of a method for using one-button shut down interrupt logic as a feature of the system in accordance with the present invention for allowing the computing device to be used as a server.

FIG. **8** is a flow chart of one embodiment of a method for a shut down interrupt routine in the system in accordance with the present invention for allowing the computing device to be used as a server.

FIG. **9** is a flow chart of one embodiment of a method for using one-button Init interrupt logic as a feature of the system in accordance with the present invention for allowing the computing device to be used as a server.

FIG. **10** is a flow chart of one embodiment of a method for an Init interrupt routine in the system in accordance with the present invention for allowing the computing device to be used as a server.

FIG. **11** is a flow chart of one embodiment of a method for using one-button power on control logic as a feature of the system in accordance with the present invention for allowing the computing device to be used as a server.

DETAILED DESCRIPTION OF THE INVENTION

The present invention relates to an improvement in computer systems. The following description is presented to enable one of ordinary skill in the art to make and use the invention and is provided in the context of a patent application and its requirements. Various modifications to the

preferred embodiment will be readily apparent to those skilled in the art and the generic principles herein may be applied to other embodiments. Thus, the present invention is not intended to be limited to the embodiment shown, but is to be accorded the widest scope consistent with the principles and features described herein.

The present invention provides a method and system for providing a modulized server on a board. The server-on-a-board is installed on a computing device. The method and system include providing bus interface logic, providing local control BIOS, flash memory and, preferably, a set of control button connectors, light emitting diodes (LED) connectors and a liquid crystal display (LCD) connector. The local control BIOS is coupled with the bus interface logic and the flash memory. The bus interface logic interacts with the computing device and allows computing device to detect the server board. The local control BIOS boots up the server and prepares the computing device for use as the server. The flash memory stores a server image for the server, which is provided to the computing device using the local control BIOS. The control button connectors allow the server to be turned on, shut down gracefully, or restored to its initial state, by a single press of buttons connected to these connectors. The LED and LCD connectors allow the system status to be displayed or shown.

The present invention will be described in terms of a particular computing device and a system having certain components. However, one of ordinary skill in the art will readily recognize that this method and system will operate effectively for other computing devices and other systems having other components performing substantially the same functions.

To more particularly illustrate the method and system in accordance with the present invention, refer now to FIG. **2**, depicting a high-level block diagram of a system **100** in accordance with the present invention for allowing the computing device to be used as a server. The system **100** is to be used in conjunction with a computing device such as the computing device **10**. The system **100** includes bus interface logic **102**, local control BIOS **104**, memory **106** and, in a preferred embodiment, other control logic **108** and connectors **109**. The components **102**, **104**, **106**, **108** and **109** of the system **100** are preferably integrated into a single board that can be plugged into the computing device **10**. The system **100** is also preferably used in conjunction with a system having a generic user interface, such as Windows 2000® operating system. The system **100** attaches to the computing device **10** via the bus interface logic **102** and bus interface **103** of the system **100** and the bus interface **26** of the computing device **10**. In operation, the computing device **10** detects the system **100** through the bus interface logic **102**, using the bus protocols of the computing device **10**. The local control BIOS **104** boots up the server and prepares the computing device for use as the server. The memory **106** includes a server image **110** for the server being provided by the system **100**. Preferably, the server image **110** is compressed and stored on the memory **106**. The server image **110** is preferably loaded onto the computing device **10** and boots up, as discussed below. Once booted up, the server image **110** allows the computing device **10** to function as a server. In addition, the system **100** also includes the other control logic **108**. In a preferred embodiment, the other control logic **108** is managed by the local control BIOS **104**. The connectors **109** preferably include an Init connector **112**, a shut-down connector **114**, a power control connector **116**, a status LED connector **118**, a DC power LED connector **120** and a LCD display connector **122**. However, in another embodiment, the other control logic **108** could include other components. The connectors **109** can be coupled to LEDs (not shown) and an LCD display (not shown) for the board. The connectors **109** are controlled using the other control logic **108**.

FIG. **3** depicts one embodiment of the local BIOS **104**. The local BIOS **104** includes a system initialization and testing block **130**, a local BIOS run-time main program **132**, an LCD display driver **134**, a memory driver **136**, a shut-down interrupt service routine **138**, and an Init service routine **140**. The drivers **134** and **136** are used to drive the display **122** and the memory **106**. The shut-down interrupt service routine **138** and Init service routine **140** are used in conjunction with the other control logic **108** described below.

Referring to FIGS. **2** and **3**, in operation, once the computing device **10** detects the presence of the system **100**, the local BIOS **104** is activated. The local BIOS **104** preferably connects with the BIOS **22** and begins controlling the computing device **10**. The local BIOS **104** preferably performs tests on the system **100** to ensure that the system **100** can control the functions of the computing device **10** as desired. For example, the local control BIOS **104** ensures that the display, memory and other input/output devices can be controlled. For example, in a preferred embodiment, the hardware identification of the flash memory **106** is read to determine the size of the memory **106**. The system initialization and testing block **130** preferably performs the testing functions. An Ethernet MAC address of the computing device **10** is also preferably read to ensure that security and personalization of the computing device **10** is preserved. In a preferred embodiment, an identification for the system **100** is read by the local control BIOS **104** to determine a version of the system **100**. The local control BIOS **104** also preferably establishes a unique personalized key, discussed below. The local control BIOS **104** establishes a boot-up sequence on the computing device **10**. The memory **106** is then mounted and boots up. The server image **110** is then extracted from the memory **106** using the unique personalized key. Without the key, the server image preferably cannot extract and utilize the server image **110**.

FIG. **4** is a diagram of one embodiment of the images for the server stored in the memory **106**. The server image **110** includes a default field configurable and field upgradeable bitmap image **141** of the other control logic **108**, an active field configurable and field upgradeable bitmap image **142** of the other control logic **108**, a default compressed server image **143**, an active server image **144**, a default flash drive boot-up image **145** and an active flash drive boot-up image **146**. The bitmaps **141** and **142** indicate the default and actual (active) bitmap images for the control logic to allow the server to track and utilize the control logic **108**. The compressed server images **143** and **144** are the default and actual (active) server images for loading onto the computing device **10**. The active server image **144** thus corresponds to the server image **110**, depicted in FIG. **2**, that is loaded onto the computing device. The flash drive images **145** and **146** are the default and actual (active) boot-up images of the flash memory **106**. Once the server image **110** is loaded on the computing device **10**, the computing device **10** can function as a server. Furthermore, the defaults can be restored, for example in an Init interrupt, described below in FIG. **10**, using the defaults **141**, **143** and **145**. The shut-down interrupt service routine **138** and Init service routine **140** can optionally reside in the server image of **110** as well.

FIG. **5** is a more detailed block diagram of one embodiment of the other control logic **108** in the system **100** in

accordance with the present invention for allowing the computing device to be used as a server. The other control logic **108** includes a local BIOS **104** address decode and control **150**, a flash memory address decode and control **152**, an LCD address decode and control **154**, one button shut-down interrupt logic **156**, ID, status and control decode **158** and one button Init interrupt logic **160**. These blocks are used to provide the additional functions, described below, such as a one button shut down and Init interrupt.

FIG. **6** is a flow chart of one embodiment of a method **200** in accordance with the present invention for using the system **100**. The method **200** preferably commences after the computing device **10** has found the system **100**. The method **200** is described in the context of the components depicted in FIGS. **1-5**. Referring to FIGS. **1-6**, the local control BIOS **104** is automatically coupled with the BIOS **22** of the computing device **10**, via step **202**. The local control BIOS **104** takes control of the computing device **10**, via step **204**. The functions of the system **100** are tested, via step **206**.

It is determined whether the test(s) performed in step **206** indicate that the system **100** is functioning properly, via step **208**. If not, then the method **200** terminates, via step **220**. If it is determined that the system **100** runs properly, then the memory **106** is mounted on the computing device **10**, via step **210**. The boot up of the computing device **10** is then performed from the memory **106** that was just mounted, via step **212**. The server image **110** is found, decompressed if necessary, via step **214**. It is determined whether the functions of the method **200** were properly performed, via step **216**. If so, then control is passed to the server, via step **218**. Otherwise, the method **200** ends at step **220**.

Thus, the method **200** and system **100** allow the computing device **10** to be used as a server. Because most of the method **200** is performed automatically, the user need not manually configure the computing device **10**. Instead, the user merely plugs in the board on which the system **100** is integrated. Thus, the process used to allow a computing device **10** to be used as a server is simplified. Moreover, the system **100** is relatively inexpensive, often costing on the order of less than $25 in quantity. Thus, the computing device **10** can be turned into a server relatively cheaply and easily.

The system **100** also preferably uses the other controls **108** and connectors **109** to provide other functions in the server. FIG. **7** depicts one embodiment of a method **220** for utilizing one button shut-down interrupt logic **156** and the shut-down connector **114**. The one button shut-down interrupt logic **156** waits for input, via step **222**. In a preferred embodiment, the input includes a push button (not shown) being depressed for a particular time. It is determined whether shut-down input was received, via step **224**. If not then step **222** is returned to. Otherwise, clock sampling is performed to allow for hardware debounce, via step **226**. It is determined whether the input was valid shut-down input, via step **228**. In a preferred embodiment, valid shut-down input includes the push button being depressed for a particular time. If the input was not valid, then step **222** is returned to. Otherwise, further shut-down interrupts are inhibited, via step **230**. Step **230** ensures that the method **220** can be completed for the valid shut down input already provided. A shut down interrupt to the server is then generated, via step **232**. A method for generating such an interrupt is described below with respect to FIG. **8**. The main system power is then shut down and the system **100** is put into stand-by mode, via step **234**. Thus, the system **100** can be shut down using a single press of a button. A user can, therefore, shut down the server provided using the system **100** relatively quickly and easily, through the use of a single button.

FIG. **8** is a flow chart of one embodiment of a method **240** for a shut down interrupt routine in the system **100** in accordance with the present invention. The method **240** is preferably implemented in conjunction with the one button shut-down interrupt logic **156**. A shut-down interrupt service routine entry is provided, via step **242**. A status port of the system **100** is read, via step **244**. The status port of the system **100** indicates whether a shut down is pending. It is determined whether a shut down is pending, via step **246**. If not, then the method **240** is terminated, via step **254**. Otherwise, a shut down sequence for the server is initiated, via step **248**. The server is then shut down, via step **250**. The main power to the system **100** is then shut down and the system **100** is put into standby mode, via step **252**. Thus, the system **100** can be shut down relatively simply and easily.

FIG. **9** is a flow chart of one embodiment of a method **260** for using one-button Init interrupt logic a feature of the system **100** in accordance with the present invention. The method **260** is used in conjunction with the one button Init interrupt logic **160** and the Init connector **112**. The one button Init interrupt logic **160** waits for connector input, via step **262**. The connector input is preferably a push button (not shown) being depressed. It is determined whether Init input is received, via step **264**. If not, step **262** is returned to. Otherwise, clock sampling is performed to allow for hardware de-bounce, via step **266**. It is determined whether the Init input received is valid, via step **268**. If not, step **262** is returned to. Otherwise, further Init interrupts are inhibited, via step **270**. Step **270** ensures that the method **260** can be completed for valid Init input already received. An Init interrupt to the server is then generated, via step **272**. The server is thus restored to its default state using the method **260**. The return to the default state is preferably found in the default server image **143** residing on the memory **106**.

FIG. **10** is a flow chart of one embodiment of a method **280** for an Init interrupt routine in the system **100** in accordance with the present invention. The method **280** is preferably used for performing the step **272** of the method **260**.

A Init interrupt service routine entry is provided, via step **282**. A status port of the system **100** is read, via step **284**. The status port of the system **100** indicates whether an initialization is pending. It is determined whether an initialization is pending, via step **286**. If not, then the method **280** is terminated, via step **290**. Otherwise, the server is restored to its default state, via step **288**. Thus, the system **100** can be initialized relatively simply and easily, by a push of a button by a user.

FIG. **11** is a flow chart of one embodiment of a method **300** for using one-button shut down and power on control logic as a feature of the system **100**. The method **300** is preferably performed using the power on control connector **116** and the shut-down connector **114**. The power control connector (not shown) of the computing device **10** is coupled with a power-on connector **116**, via step **302**. The AC power to the system **100** is then turned on, the DC power to the system **100** turned off, and the server of the system **100** placed in standby mode, via step **304**. It is determined whether the shut-down button has been depressed, via step **306**. If not, step **306** is returned to. Otherwise, DC power for the system **100** is turned on and the system **100** boots up, via step **308**. It is then determined whether power is to be disabled, via step **310**. If so, then the power on is asserted, via step **314** and the system DC power turned off via step **324**. If power is not to be disabled, then it is determined whether the shut-down interrupt is to be enabled, via step **312**. If not, it is determined whether the shut-down button

has been pressed, via step **322**. If so, then the system DC power is turned off, via step **324**. Otherwise, the method returns to step **310**. If it is determined in step **312** that the shut-down interrupt is to be enabled, power on is deasserted, via step **316**. It is then determined whether the shut-down button has been pressed, via step **318**. Preferably, step **318** determines whether the shut-down button has been pressed for a particular amount of time. If not, then the method returns to step **310**. Otherwise, the shutdown input is generated, via step **320** and step **310** returned to.

Thus, using the method **300**, the shut-down button can be used in different ways. If the shut down button is pressed prior to a shut-down interrupt being enabled, then the method **300** allows the DC power to the system **100** to be turned off. If, however, the shutdown interrupt was enabled, as determined in step **312**, prior to the shut-down button being pressed, then the shut down input generated in step **320** and the system **100** can be shut down using the method **220**. Thus, using the method **300**, the shut-down button can be used either to turn off the DC power to the system or to shut down the system **100**. Thus, using the methods **220**, **240**, **260**, **280** and **300**, additional functions can be provided using the system **100**.

A method and system has been disclosed for allowing a computing device to be used as a server. Software written according to the present invention is to be stored in some form of computer-readable medium, such as memory, CD-ROM or transmitted over a network, and executed by a processor. Consequently, a computer-readable medium is intended to include a computer readable signal which, for example, may be transmitted over a network. Although the present invention has been described in accordance with the embodiments shown, one of ordinary skill in the art will readily recognize that there could be variations to the embodiments and those variations would be within the spirit and scope of the present invention. Accordingly, many modifications may be made by one of ordinary skill in the art without departing from the spirit and scope of the appended claims.

What is claimed is:

1. A system for providing a server-on-a-board on a computing device, the computing device including at least a processor and an optional mass storage device, the system comprising:

bus interface logic for interfacing between the computing device and the system, the bus interface logic allowing the computing device to detect the system;

a local control BIOS coupled with the bus interface logic, the local control BIOS for booting up the server and preparing the computing device for use as the server; and

a memory for storing a server image for the server, the server image being provided to the computing device using the local control BIOS.

2. The system of claim **1** further comprising:

a plurality of control button connectors;

a plurality of buttons, the plurality of control button connectors for allowing the server to be turned on, shut down gracefully, or restored to its initial state, by a single press of at least one of the plurality of buttons connected to the plurality of control button connectors;

a plurality of LED and LCD connectors allowing the system status to be displayed or shown.

3. The system of claim **1** wherein the memory is a flash memory.

4. The system of claim **1** further comprising:

control logic.

5. The system of claim **4** further comprising:

a push button; and

wherein the control logic further includes a one-button init connector, coupled with the push button, for restoring the server to a default state in response to the push button being depressed for a particular time.

6. The system of claim **4** further comprising:

a push button; and

wherein the control logic further includes a shut-down connector, coupled with the push button, the shut-down connector shutting down the server gracefully if the push button is pressed for a particular time.

7. The system of claim **4** wherein the control logic further includes a power-on connector; and

wherein the control logic further includes a power-on connector connecting to the power-on connector of the system board, coupled with the shut-down push button, the power-on connector further turns the power supply on if the push button is depressed when the computing device is supplied with AC power.

8. The system of claim **4** further comprising:

a light emitting diode (LED) connector; and

wherein the control logic further includes a status LED connector coupled with the LED for indicating a operating status of the system.

9. The system of claim **4** further comprising:

a light emitting diode (LED) connector; and

wherein the control logic further includes a power-on LED connector coupled with the LED for indicating a power status of the system.

10. The system of claim **4** further comprising:

a liquid crystal display (LCD) connector; and

wherein the control logic further includes a LCD display connector coupled with the LCD for indicating a operating status of the system.

11. The system of claim **1** wherein the bus interface logic, the local BIOS control logic, a flash memory and a set of control button connectors, light emitting diodes (LED) connectors and a liquid crystal display (LCD) connector are incorporated into a single board.

12. A method for providing a server-on-a-board on a computing device, the computing device including at least a processor and an optional mass storage device, the method comprising the steps of:

(a) providing a board including bus interface logic, a local control BIOS, a flash memory, the bus interface logic for interfacing between the computing device and the system, the bus interface logic allowing the computing device to detect the system, the local control BIOS coupled with the bus interface logic, the local control BIOS for booting up the server and preparing the computing device for use as the server, the memory for storing a server image for the server, the server image being provided to the computing device using the local control BIOS; and

(b) allowing a user to utilize the server access using the board.

13. The method of claim **12** wherein the board further includes a plurality of control button connectors, a plurality of light emitting diodes (LED) connectors and a liquid crystal display (LCD) connector, the plurality of control

button connectors allowing the server to be turned on, shut down gracefully, or restored to an initial state, by a single press of buttons connected to the plurality of control button connectors, the plurality of LED connectors and the LCD connector allowing the system status to be displayed or shown.

14. The method of claim 12 wherein the memory is a flash memory.

15. The method of claim 12 wherein the board further includes control logic.

16. The method of claim 15 wherein the board further includes a push button; and

wherein the control logic further includes a one-button init connector, coupled with the push button, for restoring the server to a default state in response to the push button being depressed for a particular time.

17. The method of claim 15 wherein the board further includes a push button; and

wherein the control logic further includes a shut-down connector, coupled with the push button, the shut-down connector shutting down the server gracefully if the push button is pressed for a particular time.

18. The method of claim 15 wherein the control logic further includes a power-on connector; wherein the computing device includes a system board; and

wherein the control logic further includes a power-on connector connecting to a power-on connector of the system board for the computing device, coupled with the shut-down push button, the power-on connector further turns the power supply on if the push button is depressed when the computing device is supplied with AC power.

19. The method of claim 15 further comprising the step of:

providing a light emitting diode (LED) connector; and

wherein the control logic further includes a status LED connector coupled with the LED for indicating a operating status of the system.

20. The method of claim 15 further comprising the step of:

providing a light emitting diode (LED) connector; and

wherein the control logic further includes a power-on LED connector coupled with the LED for indicating a power status of the system.

21. The method of claim 15 further comprising the step of:

providing a liquid crystal display (LCD) connector; and

wherein the control logic further includes a LCD display connector coupled with the LCD for displaying a operating status of the system.

22. The method of claim 12 wherein the bus interface logic, the local BIOS control logic, the flash memory and a set of control button connectors, light emitting diodes (LED) connectors and a liquid crystal display (LCD) connector, are incorporated into a single board.

23. A method for providing a server-on-a-board on a computing device, the computing device including at least a processor and an optional mass storage device, the method comprising the steps of:

detecting a system for providing the server using bus interface logic in the system;

accessing a local control BIOS on the system;

using the local control BIOS for preparing the computing device for use as the server and booting up the server, for accessing a memory in the system for storing a server image for the server, the server image being provided to the computing device using the local control BIOS.

24. The method of claim 23 further comprising the steps of:

using a plurality of control button connectors allowing the server to be turned on, shut down gracefully, or restored to its initial state, by a single press of buttons connected to the plurality of control button connectors;

using the LED and LCD connectors allowing the system status to be displayed or shown.

* * * * *

UNITED STATES
DISTRICT COURT
SOUTHERN DISTRICT OF CALIFORNIA
SAN DIEGO DIVISION

 # 149302 - SH
*** * C O P Y * ***
April 02, 2008
13:27:00

Civ Fil Non-Pris
USAO #.: 08CV0602
Judge..: JEFFREY T MILLER
Amount.: $350.00 CK
Check#.: BC0038989

Total-> $350.00

FROM: ASUSTEK COMPUTER INC V. INTERN
BUSINESS MACHINES CORP

JS 44 (Rev. 11/04)

CIVIL COVER SHEET

FILED

The JS 44 civil cover sheet and the information contained herein neither replace nor supplement the filing and service of pleadings or other papers as required by law, except as provided by local rules of court. This form, approved by the Judicial Conference of the United States in September 1974, is required for the use of the Clerk of Court for the purpose of initiating the civil docket sheet. (SEE INSTRUCTIONS ON THE REVERSE OF THE FORM.)

I. (a) PLAINTIFFS
ASUSTEK COMPUTER, INC.

(b) County of Residence of First Listed Plaintiff Taipei, Taiwan
(EXCEPT IN U.S. PLAINTIFF CASES)

(c) Attorney's (Firm Name, Address, and Telephone Number)
Fish & Richardson P.C., 12390 El Camino Real, San Diego, CA 92130
858-678-5070

DEFENDANTS
INTERNATIONAL BUSINESS MACHINES CORPORATION

08 APR -2 PM 1:20
CLERK, U.S. DISTRICT COURT
SOUTHERN DISTRICT OF CALIFORNIA
DEPUTY

County of Residence of First Listed Defendant
(IN U.S. PLAINTIFF CASES ONLY)
NOTE: IN LAND CONDEMNATION CASES, USE THE LOCATION OF THE LAND INVOLVED.

Attorneys (If Known)

'08 CV 602 JM WMc

II. BASIS OF JURISDICTION (Place an "X" in One Box Only)

- ☐ 1 U.S. Government Plaintiff
- ☒ 3 Federal Question (U.S. Government Not a Party)
- ☐ 2 U.S. Government Defendant
- ☐ 4 Diversity (Indicate Citizenship of Parties in Item III)

III. CITIZENSHIP OF PRINCIPAL PARTIES (Place an "X" in One Box for Plaintiff and One Box for Defendant)
(For Diversity Cases Only)

	PTF	DEF		PTF	DEF
Citizen of This State	☐ 1	☐ 1	Incorporated *or* Principal Place of Business In This State	☐ 4	☐ 4
Citizen of Another State	☐ 2	☐ 2	Incorporated *and* Principal Place of Business In Another State	☐ 5	☐ 5
Citizen or Subject of a Foreign Country	☐ 3	☐ 3	Foreign Nation	☐ 6	☐ 6

IV. NATURE OF SUIT (Place an "X" in One Box Only)

CONTRACT
- ☐ 110 Insurance
- ☐ 120 Marine
- ☐ 130 Miller Act
- ☐ 140 Negotiable Instrument
- ☐ 150 Recovery of Overpayment & Enforcement of Judgment
- ☐ 151 Medicare Act
- ☐ 152 Recovery of Defaulted Student Loans (Excl. Veterans)
- ☐ 153 Recovery of Overpayment of Veteran's Benefits
- ☐ 160 Stockholders' Suits
- ☐ 190 Other Contract
- ☐ 195 Contract Product Liability
- ☐ 196 Franchise

REAL PROPERTY
- ☐ 210 Land Condemnation
- ☐ 220 Foreclosure
- ☐ 230 Rent Lease & Ejectment
- ☐ 240 Torts to Land
- ☐ 245 Tort Product Liability
- ☐ 290 All Other Real Property

TORTS

PERSONAL INJURY
- ☐ 310 Airplane
- ☐ 315 Airplane Product Liability
- ☐ 320 Assault, Libel & Slander
- ☐ 330 Federal Employers' Liability
- ☐ 340 Marine
- ☐ 345 Marine Product Liability
- ☐ 350 Motor Vehicle
- ☐ 355 Motor Vehicle Product Liability
- ☐ 360 Other Personal Injury

PERSONAL INJURY
- ☐ 362 Personal Injury - Med. Malpractice
- ☐ 365 Personal Injury - Product Liability
- ☐ 368 Asbestos Personal Injury Product Liability

PERSONAL PROPERTY
- ☐ 370 Other Fraud
- ☐ 371 Truth in Lending
- ☐ 380 Other Personal Property Damage
- ☐ 385 Property Damage Product Liability

CIVIL RIGHTS
- ☐ 441 Voting
- ☐ 442 Employment
- ☐ 443 Housing/ Accommodations
- ☐ 444 Welfare
- ☐ 445 Amer. w/Disabilities - Employment
- ☐ 446 Amer. w/Disabilities - Other
- ☐ 440 Other Civil Rights

PRISONER PETITIONS
- ☐ 510 Motions to Vacate Sentence

Habeas Corpus:
- ☐ 530 General
- ☐ 535 Death Penalty
- ☐ 540 Mandamus & Other
- ☐ 550 Civil Rights
- ☐ 555 Prison Condition

FORFEITURE/PENALTY
- ☐ 610 Agriculture
- ☐ 620 Other Food & Drug
- ☐ 625 Drug Related Seizure of Property 21 USC 881
- ☐ 630 Liquor Laws
- ☐ 640 R.R. & Truck
- ☐ 650 Airline Regs.
- ☐ 660 Occupational Safety/Health
- ☐ 690 Other

LABOR
- ☐ 710 Fair Labor Standards Act
- ☐ 720 Labor/Mgmt. Relations
- ☐ 730 Labor/Mgmt.Reporting & Disclosure Act
- ☐ 740 Railway Labor Act
- ☐ 790 Other Labor Litigation
- ☐ 791 Empl. Ret. Inc. Security Act

BANKRUPTCY
- ☐ 422 Appeal 28 USC 158
- ☐ 423 Withdrawal 28 USC 157

PROPERTY RIGHTS
- ☐ 820 Copyrights
- ☒ 830 Patent
- ☐ 840 Trademark

SOCIAL SECURITY
- ☐ 861 HIA (1395ff)
- ☐ 862 Black Lung (923)
- ☐ 863 DIWC/DIWW (405(g))
- ☐ 864 SSID Title XVI
- ☐ 865 RSI (405(g))

FEDERAL TAX SUITS
- ☐ 870 Taxes (U.S. Plaintiff or Defendant)
- ☐ 871 IRS—Third Party 26 USC 7609

OTHER STATUTES
- ☐ 400 State Reapportionment
- ☐ 410 Antitrust
- ☐ 430 Banks and Banking
- ☐ 450 Commerce
- ☐ 460 Deportation
- ☐ 470 Racketeer Influenced and Corrupt Organizations
- ☐ 480 Consumer Credit
- ☐ 490 Cable/Sat TV
- ☐ 810 Selective Service
- ☐ 850 Securities/Commodities/ Exchange
- ☐ 875 Customer Challenge 12 USC 3410
- ☐ 890 Other Statutory Actions
- ☐ 891 Agricultural Acts
- ☐ 892 Economic Stabilization Act
- ☐ 893 Environmental Matters
- ☐ 894 Energy Allocation Act
- ☐ 895 Freedom of Information Act
- ☐ 900Appeal of Fee Determination Under Equal Access to Justice
- ☐ 950 Constitutionality of State Statutes

V. ORIGIN (Place an "X" in One Box Only)
☒ 1 Original Proceeding ☐ 2 Removed from State Court ☐ 3 Remanded from Appellate Court ☐ 4 Reinstated or Reopened ☐ 5 Transferred from another district (specify) ☐ 6 Multidistrict Litigation ☐ 7 Appeal to District Judge from Magistrate Judgment

VI. CAUSE OF ACTION
Cite the U.S. Civil Statute under which you are filing (Do not cite jurisdictional statutes unless diversity):
35 U.S.C. section 271
Brief description of cause:
Infringement of two US patents

VII. REQUESTED IN COMPLAINT:
☐ CHECK IF THIS IS A CLASS ACTION UNDER F.R.C.P. 23
DEMAND $ TBD
CHECK YES only if demanded in complaint:
JURY DEMAND: ☒ Yes ☐ No

VIII. RELATED CASE(S) IF ANY (See instructions): JUDGE _____ DOCKET NUMBER _____

DATE
4-2-08

SIGNATURE OF ATTORNEY OF RECORD

FOR OFFICE USE ONLY

RECEIPT # 149302 AMOUNT $350 APPLYING IFP _____ JUDGE _____ MAG. JUDGE _____

SEL 4/2/08

CR